Com SIS
Computer Science and Information Systems
Volume 20, Number 2, April 2023

Contents

ISSN: 2406-1018 (Online)

Computer Science and Information Systems

Vol 20, No 2, April 2023

Com SIS

Computer Science and Information Systems

Published by ComSIS Consortium

Special Issue on Machine Learning-based Decision Support Systems in IoT systems

Volume 20, Number 2
April 2023

ComSIS is an international journal published by the ComSIS Consortium

ComSIS Editorial Office:
University of Novi Sad, Faculty of Sciences,
Department of Mathematics and Informatics
Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia
Phone: +381 21 458 888; **Fax:** +381 21 6350 458
www.comsis.org; Email: comsis@uns.ac.rs

Volume 20, Number 2, 2023
Novi Sad

Computer Science and Information Systems

Special Issue: Machine Learning-based Decision Support Systems in IoT systems

ISSN: 2406-1018 (Online)

The ComSIS journal is sponsored by:

Ministry of Education, Science and Technological Development of the Republic of Serbia
http://www.mpn.gov.rs/

ComSIS

Computer Science and Information Systems

AIMS AND SCOPE

Computer Science and Information Systems (ComSIS) is an international refereed journal, published in Serbia. The objective of ComSIS is to communicate important research and development results in the areas of computer science, software engineering, and information systems.

We publish original papers of lasting value covering both theoretical foundations of computer science and commercial, industrial, or educational aspects that provide new insights into design and implementation of software and information systems. In addition to wide-scope regular issues, ComSIS also includes special issues covering specific topics in all areas of computer science and information systems.

ComSIS publishes invited and regular papers in English. Papers that pass a strict reviewing procedure are accepted for publishing. ComSIS is published semiannually.

Indexing Information

ComSIS is covered or selected for coverage in the following:

- Science Citation Index (also known as SciSearch) and Journal Citation Reports / Science Edition by Thomson Reuters, with 2021 two-year impact factor 1.170,
- Computer Science Bibliography, University of Trier (DBLP),
- EMBASE (Elsevier),
- Scopus (Elsevier),
- Summon (Serials Solutions),
- EBSCO bibliographic databases,
- IET bibliographic database Inspec,
- FIZ Karlsruhe bibliographic database io-port,
- Index of Information Systems Journals (Deakin University, Australia),
- Directory of Open Access Journals (DOAJ),
- Google Scholar,
- Journal Bibliometric Report of the Center for Evaluation in Education and Science (CEON/CEES) in cooperation with the National Library of Serbia, for the Serbian Ministry of Education and Science,
- Serbian Citation Index (SCIndeks),
- doiSerbia.

Information for Contributors

The Editors will be pleased to receive contributions from all parts of the world. An electronic version (LaTeX), or three hard-copies of the manuscript written in English, intended for publication and prepared as described in "Manuscript Requirements" (which may be downloaded from http://www.comsis.org), along with a cover letter containing the corresponding author's details should be sent to official journal e-mail.

Criteria for Acceptance

Criteria for acceptance will be appropriateness to the field of Journal, as described in the Aims and Scope, taking into account the merit of the content and presentation. The number of pages of submitted articles is limited to 20 (using the appropriate LaTeX template).

Manuscripts will be refereed in the manner customary with scientific journals before being accepted for publication.

Copyright and Use Agreement

Computer Science and Information Systems

Volume 20, Number 2, April 2023

CONTENTS

Computer Science and Information Systems 20(2): i–iii https://doi.org/10.2298/CSIS230200iC

Guest Editorial: Machine Learning-based Decision Support Systems in IoT systems

Mu-Yen Chen[1], Jose de Jesus Rubio[2], and Mirjana Ivanović[3]

[1] National Cheng Kung University, Taiwan, mychen119@gs.ncku.edu.tw
[2] Instituto Politécnico Nacional, Mexico, rubio.josedejesus@gmail.com
[3] University of Novi Sad, Faculty of Sciences, Serbia, mira@dmi.uns.ac.rs

In contemporary research in Information and Communication Technologies, the Internet of Things (IoT) is getting to be one of the most popular technologies that facilitate new interactions among things and humans to enhance the quality of life and computing systems in a lot of domains. In IoT environments, enormous amounts of data is being generated by numerous interconnected smart and intelligent devices. Many different decision-making processes rely on quality and proper processing this data to trigger events or control actuator. With the rapid development of IoT systems, Intelligent Decision Support System (IDSS) paradigm is emerging as an attractive solution for processing and managing the information of IoT devices by providing optimal Quality of Service (QoS). IDSS for IoT systems investigates the massive quantity of complex data to help industry, academicians, medical systems, smart city and other smart applications. Decision support system (DSS) is an intelligent system, which offers excellent assistant in diverse levels of information technology. Besides, it is a dynamic information model as important data is added on a uniform basis. IoT, embedded devices, sensors, mobile applications, manual data entry and online sources are few complex data sources for IDSS. IDSS makes use of a specter of powerful Machine Learning techniques to enhance the process of complex making decisions and prediction. Other Artificial Intelligence techniques such as Metaheuristic, Fuzzy Logic, Case-based Reasoning, Artificial Neural Networks, and Intelligent Agents can be integrated into DSS for IoT systems to increase their reliability and performances. Despite the importance of decision making on IoT systems, this special issue invited researchers to publish selected original papers presenting intelligent trends to solve new challenges of IDSS problems.

This special issue received 43 submissions where the corresponding authors were majorly counted by the deadline for manuscript submission with an open call-for-paper. All these submissions were considered significant in the field, but however, one-half of them passed the pre-screening and quality check by guest editors. The qualified papers then went through double-blinded peer review based on a strict and rigorous review policy of ComSIS Journal. After a totally three-round review, 16 papers were accepted for publication. A quick overview to the papers in this issue can be revealed below, and we expect the content may draw attention from public readers, and furthermore, prompt the society development.

The first article entitled “Adaptive Multiscale Sparse Unmixing for Hyperspectral Remote Sensing Image”, by Li et al. present an adaptive multi-scale spatial sparse hyperspectral unmixing algorithm (AMUA) to deal with the unmixing hyperspectral images. The results show the robustness and accuracy of the AMUA algorithm.

The second article entitled “Evaluation of Smart City Construction and Optimization of City Brand Model under Neural Networks”, by Li et al. integrate the artificial neural

network (ANN) and grey relational analysis (GRA) to evaluate the smart city construction and city brand. Finally, the results show the proposed model has the better performance than the EWM, GRA and TOPSIS methods.

The third article entitled "Security Performance Analysis of Active Intelligent Reflective Surface Assisted Wireless Communication", by Li et al. apply the Intelligent Reflecting Surface (IRS) to wireless security communication, and design passive/active IRS methods to improve and optimize the security capacity.

The fourth article entitled "Content-only attention Network for Social Recommendation", by Wu et al. develop a content-only multi-relational attention network (CMAN) for social recommendation. The proposed model jointly estimates the score similarity and trust relationship on user similarity, and further achieves the high recommendation accuracy.

The fifth article entitled "Machine Learning-based Intelligent Weather Modification Forecast in Smart City Potential Area", by Chao adopts the machine learning approach and decision tree algorithms to develop the weather prediction model. The experimental results illustrate the proposed model can outperform the existing mesoscale regional prediction methods for the bad weather variation.

The sixth article entitled "Predicting Smart Cities' Electricity Demands Using K-Means Clustering Algorithm in Smart Grid ", Wang et al. adopt the back propagation neural network (BPNN) and K-means clustering algorithm to build the electricity demands prediction model. The experimental results illustrate the proposed model can achieve high accuracy rate of 85.25% and provide a valuable reference for power industry.

The seventh article entitled "Using Machine Learning Approach to Construct the People Flow Tracking System for Smart Cities", by Yao et al. propose a human target detection algorithm based on convolutional neural network (CNN) model and then construct a people flow tracking system to detect pedestrians effectively. This research is useful for the people flow statistics in public areas under smart city issues.

The eighth article entitled "Using neural network to automatic manufacture product label in enterprise under IoT environments", by Zhang and Dong applies the KNN, SVM, decision tree, random forest, GBDT, XGBoost, and ANN methods into the manufacturing dataset. Finally, the artificial intelligence model can decrease the frequency of production line shutdown and increase the factory productivity.

The ninth article entitled "Using deep learning to automatic inspection system of printed circuit board in manufacturing industry under the internet of Things", by Zhang adopts the deep learning approach to construct the automatic optical inspection (AOI) system for printed circuit board (PCB) defects in Industry 4.0. The results show the image recognition rate is 92% and the component recognition rate reaches 99%.

The tenth article entitled "Large-scale Image Classification with Multi-perspective Deep Transfer Learning", by Wu et al. modify the channel attention module and spatial attention module to construct a multi-perspective convolutional neural network model. The results point out the proposed model outperform other machine learning and deep learning models on ImageNet-1K and Cifar-100 datasets.

In "Using artificial intelligence assistant technology to develop animation games on mobile devices", Zhang develops an XNA animation game with AI technology including the finite state machine, fuzzy state machine and neural network to improve the learning ability of AI roles than traditional rule-base system. The results demonstrate the ad-

vantages of AI can bring important features such as scalability, reusability and design flexibility for the animation games development.

In “Pedestrian attribute recognition based on dual self-attention Mechanism”, by Fan and Guan use the spatial and semantic self-attention mechanism to construct a robust pedestrian attribute recognition framework based on convolutional neural network model. As the results, the proposed model can improve the recognition accuracy of attributes in both the PETA and RAP pedestrian attribute datasets.

In “Inverse Halftoning Based on Sparse Representation with Boosted Dictionary”, Yang et al. present an image inverse halftone algorithm for error diffusion halftone image by using the deconvolution and the denoising approaches. The experimental results illustrate the proposed algorithm achieves the better PSNR than other comparison algorithms.

In “Selective Ensemble Learning Algorithm for Imbalanced Dataset”, Du et al. develop a selective ensemble learning algorithm based on under-sampling approach for imbalanced dataset. In this study, the G-mean or F-mean will be used to calculate the classification performance of the ensemble stage. As the results, the proposed algorithm obtains the good generalization capability on imbalanced dataset for both of UCI dataset and KDDCUP dataset.

In “Personalization Exercise Recommendation Framework based on Knowledge Concept Graph”, Yan et al. integrate the knowledge structure diagram and constructivist learning theory to build a personalized exercise recommendation framework. The framework is implemented based on a graph neural network and the experimental results are better than other four methods (BKT, DKT, GKT and HGKT).

The last paper “The Duration Threshold of Video Content Observation: An Experimental Investigation of Visual Perception efficiency” by Song et al. design an investigation process into the efficiency of human visual perception on video clips considering exposure duration. The proposed method adopts several indicators to evaluate the performance of image processing, which is different from the traditional AI methods.

Acknowledgments. The guest editors are thankful for the hard work and enthusiasm of our authors and reviewers without whom the current issue and the publication of the journal itself would not have been possible. We also thank the Editorial Team members Vladimir Kurbalija, Jovana Vidaković and Davorka Radaković for technically guiding the whole process of preparation of this issue.

Computer Science and Information Systems 20(2):551–572 https://doi.org/10.2298/CSIS220828009L

Adaptive Multiscale Sparse Unmixing for Hyperspectral Remote Sensing Image*

Yalan Li[1,2], Qian Du[3*] , Yixuan Li[4], Wenwu Xie[4,*], Jing Yuan[5], Shang Lin Li[1,2,*],

and Qi Chen[6]

[1] Xiangnan University, Chenzhou,China
{liyalan, lsl}@xnu.edu.cn
[2] Hunan Engineering Research Center of Advanced Embedded Computing and Intelligent Medical Systems, 423099 Chenzhou, China
{liyalan, lsl}@xnu.edu.cn
[3] Linyi University, Linyi, China
duqian@lyu.edu.cn
[4] Hunan Institute of Science and Technology, Yueyang, China
{812011120131-vip, gavinxie}@hnist.edu.cn
[5] Institute of Disaster Prevention, Beijing, China
yuanjing20110824@sina.com
[6] Hubei Normal University, Huangshi, China.
chenqi@hbnu.edu.cn

Abstract. Sparse unmixing of hyperspectral images aims to separate the endmembers and estimate the abundances of mixed pixels. This approach is the essential step for many applications involving hyperspectral images. The multi-scale spatial sparse hyperspectral unmixing algorithm (MUA) could achieve higher accuracy than many state-of-the-art algorithms. The regularization parameters, whose combinations markedly influence the unmixing accuracy, are determined by manually searching in the broad parameter space, leading to time consuming. To settle this issue, the adaptive multi-scale spatial sparse hyperspectral unmixing algorithm (AMUA) is proposed. Firstly, the MUA model is converted into a new version by using of a maximum a posteriori (MAP) system. Secondly, the theories indicating that andnorms are equivalent to Laplacian and multivariate Gaussian functions, respectively, are applied to explore the strong connections among the regularization parameters, estimated abundances and estimated noise variances. Finally, the connections are applied to update the regularization parameters adaptively in the optimization process of unmixing. Experimental results on both simulated data and real hyperspectral images show that the AMUA can substantially improve the unmixing efficiency at the cost of negligible accuracy. And a series of sensitive experiments were undertook to verify the robustness of the AMUA algorithm.

Keywords: adaptive multiscale sparse hyperspectral unmixing algorithm, loss functions, regularization parameters, *maximum a posteriori*.

* Corresponding authors

1. Introduction

Hyperspectral remote sensing image (HRSI) is the data obtained by hyperspectral resolution sensors, which has led to extensive exploration in geology, medicine, aerial surveillance and other fields[1-7]. Numerous mixed pixels exist due to the low spatial resolution of the HRSI, which hinders the application of hyperspectral data. One of the key issues of HRSI systems lies in the separation of endmembers and the corresponding abundances from the mixed pixel[8-9]. This process is recognized as hyperspectral unmixing, which relies on a basic assumption of Linear Mixed Model (LMM); that is, the spectrum of a specific pixel is a linear mixture of the endmembers[8,10,11]. Sparse unmixing has attracted extensive recognition in the past few years with the increasing availability of spectral libraries. Sparse unmixing aims to estimate an abundance when the HRSI and spectral library are given in [12-19]. Thus, sparse unmixing usually is usually regarded as an ill-posed inverse problem, which indicates the absence of any unique solution. To alleviate such problems, a popular technique is widely used by adding regularizations to the loss function. Iordache et al.[20] considered that a pixel contains only a few numbers of components and introduce sparsity through the abundance map. In [20], the regularization was added to the abundance matrix to develop the sparse unmixing algorithm via variable splitting and augmented Lagrangian (SUnSAL). The SUnSAL has improved unmixing performance whilst ignoring the rich spatial correlation information among neighbouring pixels. For exploring spatial prior information, several sparse unmixing algorithms with the spatial prior information, such as SUnSAL with total variation (SUnSAL-TV) [21], have been introduced. The SUnSAL-TV utilized total variation regularization to estimate an abundance map, which has piecewise transitions at the cost of smoothing sharp discontinuities between neighbouring pixels. A new spectral – spatial weighted sparse unmixing (S2WSU) was developed to capture considerably sharp discontinuities[22]. S2WSU employed the spectral and spatial regularizations simultaneously to derive a sparse solution and obtain better abundance results[22]. However, this approach requires a considerable amount of time and is sensitive to noise. To overcome the limitations, Borsoi et al.[23] made use of multi-scale spatial information to develop a novel multi-scale sparse unmixing algorithm (MUA). In MUA, the original HRSI(in the original image domain) was transformed to a coarse-scale (CS) representation via means of segmentation techniques[24-28]. At this scale, the SUnSAL was used to estimate an initial abundance value. The initial abundance was then converted back to the original image domain to yield a low-level image, which is regarded as a novel information of the spatial context. A SUnSAL loss function was constructed in the original image domain. The new regularization term for the low-level image was added to this SUnSAL model to construct a novel unmixing mathematical model for the original HRSI. Therefore, MUA is the linear regression problem involving three regularization parameters. In the process of deducing the solutions, several existing works have reported that the estimated accuracy relies heavily on a proper configuration of these regularization parameters [29-34]. Given the desirable values, the MUA has achieved remarkable success in obtaining piecewise homogeneous abundances and sharp discontinuities among neighbouring pixels [23]. Unfortunately, the optimal regularization parameters in MUA are obtained by exhaustive search, which is unsuitable for real applications.

A new format based on the MAP estimation technique is proposed for the solution of the regularization parameter selection problem for MUA. The MAP mainly aims to

identify the unknown fractional abundances including the regularization parameters by using the Bayes rule. As an effective adaptive parameter estimation method, the MAP system is used widely used for adaptively selecting regularization parameters in deep learning networks and image super-resolution [35-40]. To build the adaptive version of MUA, the MAP is applied to redesign the unmixing problem into a unified model to obtain the relations between the optimal regularization parameters and the fractional abundances.

Inspired by the idea in [35-37], an adaptive regularization parameters strategy for MUA is introduced, which is shown in Fig. 1.

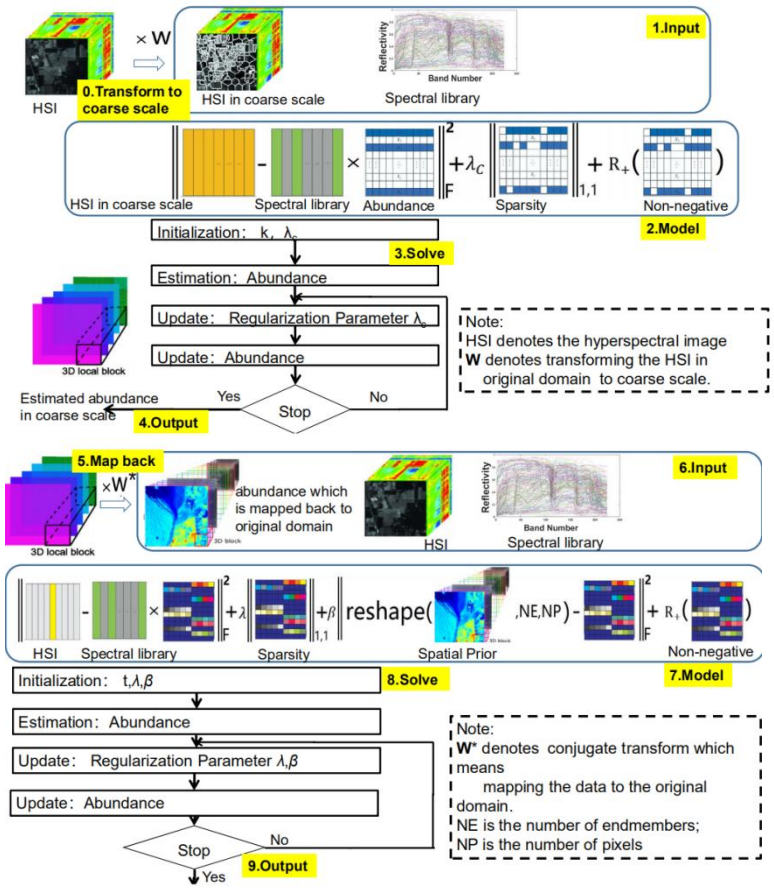


Fig.1. Framework of our proposed adaptive mutliscale spatial sparse unmixing scheme.

There are total 10 steps. At step 0, the HSI image is transformed to the coarse scale by W; At step 1, the HSI is inputted into the coarse scale and spectral libraries; At step 2, the model for unmixing the HSI in the coarse scale is constructed; At step 3, the model with adaptive regularization parameters scheme is solved; At step 4, the solution in coarse scale is obtained; At step 5, the solution by step 4 is mapped into the original domain; At step 6, the solution by step 4 is inputted to HSI and spectral libraries; At step 7, the model for unmixing with spatial prior is constructed; At step 8, the model by

step 7 with adaptive regularization parameters scheme is solved; At step 9, the final estimated abundance by automatic version of the multiscale spatial sparse unmixing algorithm is obtained. The major contributions are as follows.

1. The adaptive multi-scale spatial sparse unmixing strategy for MUA (AMUA) can adaptively choose the regularization parameters at each iteration in unmixing process.
2. The adaptive multi-scale spatial sparse unmixing strategy for MUA (AMUA) can adaptively choose the regularization parameters at each iteration in unmixing process.
3. The adaptive multi-scale spatial sparse unmixing strategy for MUA (AMUA) can adaptively choose the regularization parameters at each iteration in unmixing process.

The whole work is arranged as follows. The part 2 reviews the MUA method. The part 3 introduces the development process of the AMUA method in detail. The part 4 introduces numerous experiments and detailed comparisons. The part 5 discusses the sensitivity of the proposed method. Finally, the part 6 is the conclusion of the work.

2. Review of Multi-scale Spatial Hyperspectral Unmixing

The section describes the framework of the multiscale sparse unmixing algorithm (MUA), which is closely related to the proposed method.

Let $Y \in \mathbb{R}^{L\times N}$ have L bands and N pixels, which is HRSI. The mixing process is modelled by the linear mixing model (LMM) [8], which is given as follows:

$$\mathbf{Y} = \mathbf{AX} + \mathbf{E} \quad \text{s.t.}: \mathbf{X} \geq \mathbf{0}, 1_P^T\mathbf{X} = 1_N^T , \tag{1}$$

where $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_P] \in \mathbb{R}^{L\times P}$ represents an endmember spectral library with P endmembers; $\mathbf{X}$ is an abundance matrix and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbb{R}^{P\times N}$. $\mathbf{X} \geq \mathbf{0}$ and $1_P^T\mathbf{X} = 1_N^T$ is the abundance constraints (ANCs)[8]. Notation $[\cdot]^T$ represents vector/matrix transpose, and $\mathbf{E} \in \mathbb{R}^{L\times N}$ denotes an additive noise.

Only a few endmembers from library $\mathbf{A}$ are present in a pixel. Thus matrix $\mathbf{X}$ exhibits sparse property, and sparse unmixing can be specified as a function

$$\widehat{\mathbf{X}} = \text{argmin}_{\mathbf{X}} \frac{1}{2}||\mathbf{Y} - \mathbf{AX}||_F^2 + \lambda||\mathbf{X}||_{1,1} + R_+(\mathbf{X}) , \tag{2}$$

where $||\mathbf{X}||_{1,1}$ is the sparse regularization term and $R_+(\mathbf{X})$ represents non-negativity.

Therefore, the optimization problem of the MUA model can be written as follows [23]:

$$\widehat{\mathbf{X}_C} = \text{argmin}_{\mathbf{X}_C} \frac{1}{2}||\mathbf{Y}_C - \mathbf{AX}_C||_F^2 + \lambda_C||\mathbf{X}_C||_{1,1} , \tag{3}$$

$$\widehat{\mathbf{X}} = \underset{X>0}{\text{argmin}}||\mathbf{Y} - \mathbf{AX}||_F^2 + \lambda||\mathbf{X}||_{1,1} + \frac{\beta}{2}||\widehat{\mathbf{X}_D} - \mathbf{X}||_F^2 , \tag{4}$$

where $\mathbf{W} \in \mathbb{R}^{N\times K}$ represents a transformation; $\mathbf{W}^* \in \mathbb{R}^{K\times N}$ denotes conjugate transform; $\mathbf{Y}_C = \mathbf{YW}$ represents the transformation of the original image $\mathbf{Y}$ to coarse approximations. $\mathbf{X}_C = \mathbf{XW}$ means transforming the abundance matrix $\mathbf{X}$ to the corresponding coarse scale. $\widehat{\mathbf{X}_C}$ is the abundance estimated by equation (3) in coarse

scale and $\widehat{\mathbf{X}_D} = \widehat{\mathbf{X}_C}\mathbf{W}^*$ stands for mapping the $\widehat{\mathbf{X}_C}$ back to the original image domain to yield the coarse abundance $\widehat{\mathbf{X}_D}$. Lastly, $\widehat{\mathbf{X}}$ is the final solution to MUA [23]. λ, β and λc are the regularisation parameters.

3. Adaptive Multi-scale Spatial Sparse Hyperspectral Unmixing

This section describes the use of MAP to investigate the relationships among the estimated abundance, regularization parameters and the noise variance of the HRSI. These relationships are applied to the alternating direction multiplier (ADMM) algorithm [41] to settle the problem of adaptive multiscale spatial sparse unmixing.

The multi-scale spatial sparse unmixing comprises two mathematical models in (3) and (4). Firstly, the model in (4) is investigated: given **Y** and **A**, **X** is estimated together with the parameters λ and β. Under the Bayesian framework and MAP, the unmixing model in (4) can be reconsidered as estimating the λ, β and **X** at the same level and the problem can be rewritten as:

$$(\widehat{\mathbf{X}},\lambda,\beta)= \underset{X>0}{\operatorname{argmax}}(p(\mathbf{X},\lambda,\beta|\mathbf{Y})) . \tag{5}$$

Using Bayes' rule, (5) can be expressed as:

$$(\widehat{\mathbf{X}},\lambda,\beta) = \underset{(\mathbf{X},\lambda,\beta)}{\operatorname{argmax}}\left(\frac{p(\mathbf{Y}|\mathbf{X},\lambda,\beta)p(\mathbf{X}|\lambda,\beta)p(\lambda,\beta)}{p(\boldsymbol{Y})}\right) . \tag{6}$$

The λ, β and **X** are independent of p(**Y**); thus (6) is equivalent to (7):

$$(\widehat{\mathbf{X}},\lambda,\beta) = \underset{(\mathbf{X},\lambda,\beta)}{\operatorname{argmax}}(p(\mathbf{Y}|\mathbf{X},\lambda,\beta)p(\mathbf{X}|\lambda,\beta)p(\lambda,\beta) , \tag{7}$$

where $(p(\mathbf{Y}|\mathbf{X},\lambda,\beta)$is the likelihood distribution of the **X** equal to p(**Y**|**X**). Suppose **Y** is polluted by zero-mean white Gaussian noise, the probability distribution of p(**Y**|**X**) is as follows:

$$P(\mathbf{Y}|\mathbf{X}) = \frac{1}{\sqrt{2\pi}\delta_n}\exp\left(-\frac{||\mathbf{Y}-\mathbf{A}\mathbf{X}||_F^2}{2\delta_n^2}\right), \tag{8}$$

where δ_n is the standard deviation of the Gaussian noise.

For the abundance matrix **X** with sparse property and spatial prior information, Borsoi et al. indicated that $||\mathbf{X}||_{1,1}$ corresponds to a Laplacian probability with zero-mean [22]. Wang et al. reported that $||\widehat{\mathbf{X}_D} - \mathbf{X}||_F^2$ can be regarded as an independent and identically distributed multivariate Gaussian prior with a zero-mean value [12,42]. Therefore, the prior density P (**X**) could be written as

$$\begin{aligned} P(\mathbf{X}) &= \frac{1}{C}\exp\left(-\frac{\sqrt{2}||\mathbf{X}||_{1,1}}{\delta_x}\right)\cdot \exp\left(-\frac{||\widehat{\mathbf{X}_D} - \mathbf{X}||_F^2}{2\delta_\beta^2}\right) \\ &= \frac{1}{C}\exp\left(-\frac{\sqrt{2}||\mathbf{X}||_{1,1}}{\delta_x} - \frac{||\widehat{\mathbf{X}_D}-\mathbf{X}||_F^2}{2\delta_\beta^2}\right), \end{aligned} \tag{9}$$

where C is the normalization factor related to the sum of the succeeding exponential function and δ_β stands for the standard deviation of the map acquired by the $\widehat{\mathbf{X}_D} - \mathbf{X}$. δ_x is the standard deviation of the abundance X.

The prior for $p(\lambda, \beta)$ is a uniform distribution [37]. Substituting (8) and (9) in (7) yields

$$(\widehat{\mathbf{X}},\lambda,\beta)= \underset{(\mathbf{X},\lambda,\beta)}{\operatorname{argmax}}(log\mathrm{p}(\mathbf{Y}|\mathbf{X},\lambda,\beta) + log\mathrm{p}(\mathbf{X}|\lambda,\beta) + log\mathrm{p}(\lambda,\beta))$$

$$=\underset{(\mathbf{X},\lambda,\beta)}{\operatorname{argmax}}(-\frac{||\mathbf{Y}-\mathbf{AX}|\ |_{\mathrm{F}}^2}{2\delta_{\mathrm{n}}^2}-\frac{\sqrt{2}||\mathbf{X}||_{1,1}}{\delta_{\mathbf{x}}}-\frac{||\widehat{\mathbf{X}_{\mathrm{D}}}-\mathbf{X}|\ |_{\mathrm{F}}^2}{2\delta_{\beta}^2}+log\mathrm{p}(\lambda,\beta)+log\mathrm{C1}+log\mathrm{C2})$$

$$=\underset{(\mathbf{X},\lambda,\beta)}{\operatorname{argmin}}(\frac{||\mathbf{Y}-\mathbf{AX}||_{\mathrm{F}}^2}{2\delta_{\mathrm{n}}^2}+\frac{\sqrt{2}||\mathbf{X}||_{1,1}}{\delta_{\mathbf{x}}}+\frac{||\widehat{\mathbf{X}_{\mathrm{D}}}-\mathbf{X}||_{\mathrm{F}}^2}{2\delta_{\beta}^2}), \tag{10}$$

where logC1 is the irrelevant term from $\mathrm{p}(\mathrm{X}|\lambda,\beta)$ and $\mathrm{p}(\mathrm{Y}|\mathrm{X},\lambda,\beta)$. C2 is the standardized constant related to the stable prior value of $\mathrm{p}(\lambda,\beta)$.

From (4) and (10), the relationships between the regularization parameters λ, β and the unknown abundance $\mathbf{X}$ can be determined by the following:

$$\lambda=\frac{\sqrt{2}\delta_{\mathrm{n}}^2}{\delta_{\mathbf{x}}}\ ,\beta=\frac{\delta_{\mathrm{n}}^2}{\delta_{\beta}^2}. \tag{11}$$

Similarly, the above processing is performed on (3) to obtain the following:

$$\lambda_{\mathrm{C}}=\sqrt{2}\delta_{\mathbf{Y}_{\mathrm{C}}}^2/\delta_{\mathbf{X}_{\mathrm{C}}}\ . \tag{12}$$

where $\delta_{\mathbf{X}_{\mathrm{C}}}$ and $\delta_{\mathbf{Y}_{\mathrm{C}}}$ are the standard deviations of the abundance $\mathbf{X}_{\mathrm{C}}$ and image $\mathbf{Y}_{\mathrm{C}}$, respectively.

Table 1. Algorithm 1: AMUA

Algorithm 1: AMUA
1. **Input:** $\mathbf{A};\mathbf{Y}_{\mathrm{C}}$; set k = 1; $\mathbf{X}_{\mathrm{c}}^{(0)}=(\mathbf{A}^{\mathrm{T}}\mathbf{A}+\mathbf{3I})^{-\mathbf{1}}(\mathbf{A}^{\mathrm{T}}\mathbf{Y}_{\mathrm{C}})$; Max_Iter=50; choose $\lambda_{\mathrm{c}}^{(0)}>0$ randomly. **Repeat** $\mathbf{X}_{\mathrm{c}}^{(\mathrm{k})}=\operatorname{argmin}_{\mathbf{X}_{\mathrm{C}}}\frac{1}{2}||\mathbf{Y}_{\mathrm{C}}-\mathbf{AX}_{\mathrm{C}}||_{\mathrm{F}}^2+\lambda_{\mathrm{c}}^{(\mathrm{k}-1)}||\mathbf{X}_{\mathrm{C}}||_{1,1}$; $\delta_{\mathbf{Y}_{\mathrm{C}}}^{(\mathrm{k})}=\mathrm{std}(\mathbf{Y}_{\mathrm{C}}-\mathbf{AX}_{\mathrm{c}}^{(\mathrm{k})}))$; $\delta_{\mathbf{X}_{\mathrm{C}}}^{(\mathrm{k})}=\mathrm{std}(\mathbf{X}_{\mathrm{c}}^{(\mathrm{k})})$; $\lambda_{\mathrm{c}}^{(\mathrm{k})}=\frac{\sqrt{2}(\delta_{\mathbf{Y}_{\mathrm{C}}}^{(\mathrm{k})})^2}{\delta_{\mathbf{X}_{\mathrm{C}}}^{(\mathrm{k})}}$; **Until:** $\mathrm{abs}(\lambda_c^{(k)}-\lambda_c^{(k-1)})\le 1\mathrm{e}-6$ or $\mathrm{k}\ge$ Max_Iter **Output:** $\widehat{\mathbf{X}_{\mathrm{C}}}=\mathbf{X}_{\mathrm{c}}^{(\mathrm{k})}$
2. Input: A;Y; $\widehat{\mathbf{X}_{\mathrm{D}}}=\widehat{\mathbf{X}_{\mathrm{C}}}\mathbf{W}^{*}$; set $t=1$; Max_Iter = 50; choose $\lambda^0,\beta^0>0$ randomly;
Repeat:
$\widehat{\mathbf{X}}^{(\mathrm{t})}=\underset{\mathbf{X}>0}{\operatorname{argmin}}\frac{1}{2}\|\mathbf{Y}-\mathbf{AX}\|_{\mathrm{F}}^2+\lambda^{(\mathrm{t}-1)}\|\mathbf{X}\|_{1,1}+\frac{\beta^{(\mathrm{t}-1)}}{2}\|\widehat{\mathbf{X}_{\mathbf{D}}}-\mathbf{X}\|_{\mathrm{F}}^2$
$\delta_{\mathrm{n}}^{(\mathrm{t})}=\mathrm{std}(\mathbf{Y}-\mathbf{A}\widehat{\mathbf{X}}^{(\mathrm{t})});\delta_{\mathbf{X}}^{(\mathrm{t})}=\mathrm{std}(\widehat{\mathbf{X}}^{(\mathrm{t})})$
$\lambda^{(\mathrm{t})}=\frac{\sqrt{2}(\delta_n^{(\mathrm{t})})^2}{\delta_{\mathrm{x}}^{(\mathrm{t})}}$; $\delta_\beta^{(\mathrm{t})}=\mathrm{std}(\widehat{\mathbf{X}}^{(\mathrm{t})}-\widehat{\mathbf{X}_{\mathbf{D}}})$; $\beta^{(\mathrm{t})}=\frac{(\delta_{\mathrm{n}}^{(\mathrm{t})})^2}{\delta_\beta^{(\mathrm{t})}}$;
Until: $\mathrm{abs}(\lambda^{(t)}-\lambda^{(t-1)})\le 1e-6$; and; $\mathrm{abs}(\beta^{(t)}-\beta^{(t-1)})\le 1e-6$; or $t\ge$ MaxIter
Output: $\widehat{\mathbf{X}}=\widehat{\mathbf{X}}(\mathrm{t})$

Lastly, (13) lists the summary of regularization parameters.

$$\left.\begin{aligned}\lambda &= \frac{\sqrt{2}\delta_{\mathrm{n}}^2}{\delta_{\mathbf{x}}}\\ \beta &= \frac{\delta_{\mathrm{n}}^2}{\delta_{\beta}^2}\\ \lambda_C &= \frac{\sqrt{2}\delta_{\mathbf{Y_C}}^2}{\delta_{\mathbf{X_C}}}\end{aligned}\right\}. \tag{13}$$

Observing (13), strong relationships are found between the regularization parameters and the standard deviations (δ_{n}, δ_{x}, δ_{β}, $\delta_{\mathbf{Y_C}}$, $\delta_{\mathbf{X_C}}$). These standard deviations are unknown in the real HRS and are usually estimated as follows: δ_{n} is estimated by the method in [43], $\delta_{\mathbf{x}}$ and $\delta_{\mathbf{X_C}}$result from the abundance estimated at the corresponding iteration. The details will be described in the Algorithm 1.

The main iterative process of AMUA is listed in Algorithm 1 using the (ADMM) strategy [41], which can degrade the complex problem into a series of simple ones and efficiently solve AMUA. Notably, the transformation **W**, which was mentioned after Equation (4), appears in Algorithm 1. The **W** denotes the operation of transforming the original image to a coarser scale [23], such as a segmentation with binary partition tree (BPT) [44] and over-segmentation with the simple linear iterative clustering (SLIC)[45]. Additional details will be presented in section 4.

4. Experiments

In this part, numerous experiments were performed on two simulated datasets (DC1 and DC2), and one real HRSI to illustrate the accuracy and effectiveness of the proposed algorithms. The proposed method AMUA has two alternatives: AMUA using BPT segmentation (denoted as AMUA (BPT)) and that using SLIC segmentation (denoted as AMUA (SLIC)). The proposed methods are compared with MUA using BPT (denoted as MUA(BPT)) [23], MUA using SLIC (MUA(SLIC)) [23], SUnSAL [20], SUnSAL-TV [21] and S2WSU [22].

The optimal regularization parameters(λ, β, λ_C) are manually selected by performing a grid search for different datasets in SUnSAL, SUnSAL-TV, S2WSU and MUA algorithms. For the different datasets, Table 2 lists the optimal parameters given in [20], [21], [22], [23]. Notably, the regularization parameters(λ, β, λ_C) in AMUA algorithms are automatically estimated by an alternating iterative process. Hence, the parameters in AMUA algorithm are not constant and denoted by NA in Table 2.

Table 2. Parameters

Algorithms / Data	DC1 data cube 20dB	DC1 data cube 30dB	DC2 data cube 20dB	DC2 data cube 30dB	Curprit
SUnSAL	$\lambda = 0.7$	$\lambda = 0.1$	$\lambda = 0.1$	$\lambda = 0.01$	$\lambda = 0.01$
SUnSAL-TV	$\lambda = 0.05$	$\lambda = 0.007$	$\lambda = 0.01$	$\lambda = 0.005$	$\lambda = 0.001$
	$\lambda_{TV} = 0.05$	$\lambda_{TV} = 0.01$	$\lambda_{TV} = 0.03$	$\lambda_{TV} = 0.007$	$\lambda_{TV} = 0.001$
S2WSU	$\lambda = 0.1$	$\lambda = 0.005$	$\lambda = 0.01$	$\lambda = 0.01$	$\lambda = 7E(-5)$
MUA(BPT)	$\lambda = 0.1$	$\lambda = 0.1$	$\lambda = 0.1$	$\lambda = 0.05$	$\lambda = 0.001$
	$\lambda_C = 0.005$	$\lambda_C = 0.005$	$\lambda_C = 0.005$	$\lambda_C = 0.001$	$\lambda_C = 0.001$
	$\beta = 30$	$\beta = 30$	$\beta = 5$	$\beta = 1$	$\beta = 3$
AMUA(BPT)	**N**A	**N**A	**N**A	**N**A	**N**A
MUA(SLIC)	$\lambda = 0.1$	$\lambda = 0.05$	$\lambda = 0.1$	$\lambda = 0.03$	$\lambda = 0.001$
	$\lambda C = 0.03$	$\lambda C = 0.007$	$\lambda C = 0.007$	$\lambda C = 0.003$	$\lambda C = 0.001$
	$\beta = 30$	$\beta = 10$	$\beta = 10$	$\beta = 3$	$\beta = 30$
AMUA(SLIC)	**N**A	**N**A	**N**A	**N**A	**N**A

4.1. Simulated and real datasets for efficiency comparison

To illustrate the efficiency, Table 3 lists the running times of SUnSAL, SUnSAL-TV, S2WSU, MUA and AMUA for DC1, DC2 and Cuprite data. All the algorithms were carried out on PC, using MATLAB R2019b, an Intel Core i5-8250u CPU (1.6 GHz) and 8.00 GB RAM. For SUnSAL, SUnSAL-TV, S2WSU and MUA, the regularization parameters (λ, β, λ_C) are computed by a loop running many times with different combinations of some parameters until the best unmixing result is obtained. In most cases, the total number combinations of the parameters (TNCP) amounts to 100 at least. And for one combination, the executing time (ET) is different for different algorithms. Hence, the running time is computed by multiplying TNCP with ET. Since the parameters are estimated automatically in AMUA, there are not many parameter combinations, so the TNCP is equal to one. Compared with all other algorithms in Table 3, the AMUA efficiency is greatly improved and the total running time is greatly reduced. For DC2 instance, the total running times for SUnSAL, SUnSAL-TV, S2WSU, MUA(BPT), MUA(SLIC), AMUA(BPT) and AMUA(SLIC) are 1046.3s, 17108s, 9130.5s, 1121.7s, 650.6s, 160.496s and 74.021s. The greatly improved efficiency mainly result from that the AMUA avoids extensive search for optimal parameters.

Table 3. Running times (Simulated database for accuracy comparison)

Data / Algorithms	SUnSAL	SUnSAL-TV	S2WSU	MUA (BPT)	MUA (SLIC)	AMUA (BPT)	AMUA (SLIC)
DC1	5.187s×100	164.839s×100	50.063s×100	14.947s×100	7.164s×100	23.028s×1	19.869s×1
DC2	10.463s×100	171.08s×100	91.305s×100	11.217s×100	6.506s×100	160.496s×1	74.021s×1
Real mage	368.7s×100	2290.4s×100	961.8s×100	157.9s×100	202.5s×100	1107.8ss×1	997.34s×1

4.2. Simulated database for accuarcy comparison

Data cube DC1, which comprises 5 endmembers, has 75×75 pixels[20]. Meanwhile, data cube DC2 containing piecewise smooth abundance maps (with steep transitions) has 9 endmembers and 100×100 pixels[23]. In DC1 and DC2, the observed HRSIs were contaminated by white Gaussian noise, with signal-to-noise ratios (SNR) of 20 and 30dB[23]. To assess the accuracy of unmixing, the signal-to-reconstruction error (SRE)[21] is defined as shown below:

$$\mathrm{SRE} = 10\log_{10}\mathbb{E}\{||\mathbf{X}||_{\mathrm{F}}^2\} / \mathbb{E}\{||\widehat{\mathbf{X}} - \mathbf{X}||_{\mathrm{F}}^2\}, \quad (13)$$

where $\mathbf{X}$ and $\widehat{\mathbf{X}}$ are the true and estimated abundance, respectively; a high SRE value for the estimated abundance leads to high estimated accuracy.

Table 4 lists the SRE values of different algorithms for the data DC1 and DC2. And Fig.2 shows these values directly in a histogram. Compared with MUA, the SRE value of AMUA is slightly lower, near to that of MUA. For DC1 with a 20 dB instance, the SRE value of AMUA (BPT) achieves 12.715, the SRE of MUA(BPT) is 13.393, and the difference is only approximately 0.68. Meanwhile, the results of the AMUA algorithm were significantly better than those of SUnSAL(4.541), SUnSAL-TV(9.424) and S2WSU(7.697).

Table 4. SRE values of different algorithms on DCI and DC2.

Algorithms / Data	DC1 data cube		DC2 data cube	
	20dB	30dB	20dB	30dB
SUnSAL	4.541	8.911	4.281	10.428
SUnSAL-TV	9.424	14.441	11.554	17.988
S2WSU	7.697	15.487	9.332	21.668
MUA(BPT)	13.393	**18.257**	**13.928**	16.963

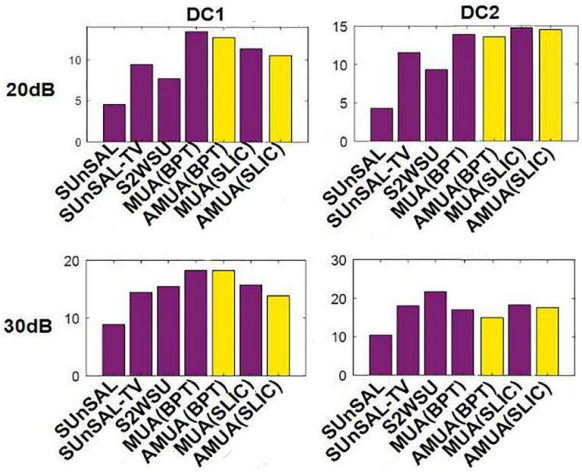


Fig. 2. SRE values of different algorithms on DC1 and DC2 by histogram

Compared with S2WSU, the SRE values of AMUA are more stable. The SRE values of S2WSU for DC2 with 30 dB instance achieve the best performance (21.668) among

listed algorithms. However, the SRE value of S2WSU degrades rapidly to 9.332 when the SNR is decreased to 20 dB. The SRE values of AMUA(BPT) for DC2 30 and 20 dB are 16.963 and 13.928, respectively. The SRE values of AMUA(SLIC) for DC2 30 and 20 dB are 17.60 and 14.488, respectively, demonstrating a stable performance.

Figs. 3 and 4 show abundance map samples from different algorithms for the second endmember of DC1 and DC2, respectively, with the upper row 20 DB SNR and the lower row 30 DB SNR.

Fig. 3 shows that the global differences between MUA and AMUA are too small to be observed, especially those between MUA(BPT) and AMUA(BPT), as shown in the red rectangles. Fig. 4 shows that compared with MUA, AMUA can improve the sharpness of edge regions between adjacent regions, such as green ellipse regions, and obtain more smooth homogeneous areas, such as the region in the yellow circle.

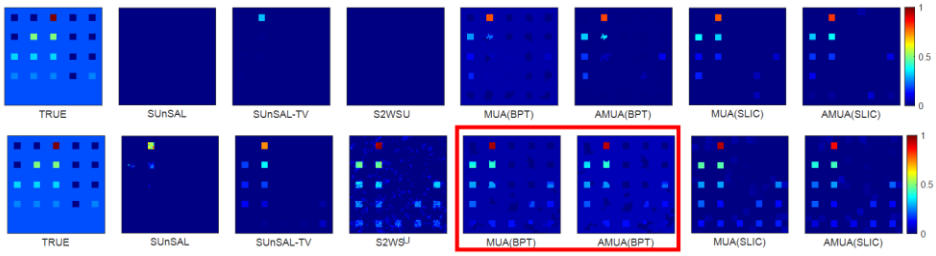


Fig. 3. Abundance maps obtained by different unmixing algorithms for the second endmember in DC1

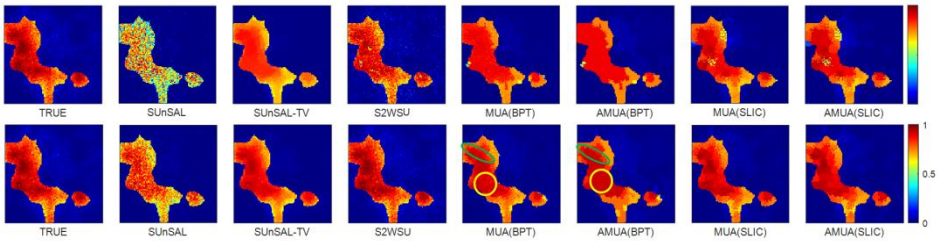


Fig. 4. Abundance maps obtained by different unmixing algorithms for the second endmember in DC2

4.3. Real hyperspectral image for accuracy comparison

To evaluate the robustness of the proposed AMUA algorithm, qualitative comparisons were constructed using Cuprite, a well-known and widely used real data set provided in [43]. The abundance images estimated by different algorithms for the minerals alunite, buddingtonite and chalcedony are shown in Fig. 5, arranged from top to bottom. The AMUA algorithms can obtain similar abundance to that of MUA algorithms. However, some parts from AMUA (BPT) are smoother than that of MUA (BPT), such as the parts highlighted by the ellipses in Fig. 5. The edge parts by AMUA (BPT) are also sharper than by MUA (BPT), such as the areas highlighted by the rectangle in Fig. 5.

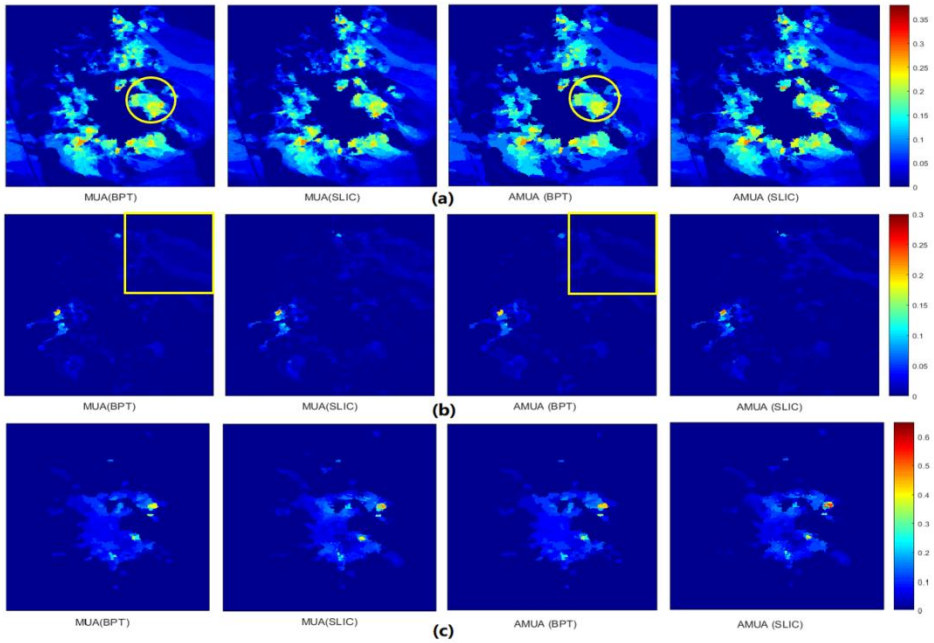


Fig. 5. Abundance maps of different algorithms methods for the data Cuprite.

5. Sensitivity discussion

This section will analyse the sensitivity of the AMUA algorithm to the regularization parameters (λ, β, λ_C), the estimated noise and the estimated abundance.

5.1. Sensitivity to regularization parameters

In the MUA, three regularization parameters(λ, β, λ_C) are manually selected to trade off the original image sparsity (λ), coarse scale sparsity (λ_C) and multiscale spatial information (β). The determination of the regularization parameters (λ,β,λ_C) greatly affects the unmixing performance. The regularization parameters($\lambda, \beta, \lambda_C$) are automatically determined in AMUA.The impact of λ, β, and λ_C on the accuracy of unmixing results with DC1 20 and 30 dB by using MUA and AMUA algorithms are shown in Figs. 6 and 7, respectively. The 4-D colour-maps illustrate the SRE values (corresponding to the bubble colour) at different combinations of λ, β and λ_C. For example, in the Fig.7 (a), the bubble A exhibits shallow blue, which denotes the SRE as 12.321 at a combination of λ=0,β=1000 and λ_C=0. The values of the regularization parameters are searched within {0,1e-15,1e-13,1e-11,1e-9,1e-7,1e-5, 1e-3 ,1e-1, 1e0, 1e1 ,1e2 ,1e3}. The sub-figures (a) and (c) in Figs. 6 and 7 illustrate the searching details in the MUA (BPT) and MUA (SLIC) algorithms, respectively. The combinations of λ, β and λ_C are automatically determined in AMUA algorithms. And the sub-figures

(b) and (d) display the performance of the AMUA (BPT) and AMUA (SLIC) algorithms, respectively.

The following conclusions are presented on the basis of Figs. 6 and 7.

(1) For the MUA, satisfactory accuracy can be achieved by carefully selecting appropriate combinations of the regularization parameters. For example, in Fig.6 (a), the highest SRE value (approximately 12.627 dB) occurs at λ = 1e-3, β = 1 and λ_C =1e-1; however, poor accuracy, such as the SRE value(-6.50 dB), occurs at improper combinations of regularization parameters. The AMUA not only obtains the acceptable accuracy, but also avoids searching the regularization parameters manually.

(2) The optimal regularization parameters vary for different datasets; such a variation would be a burden for manually searching the regularization parameters in practice. For MUA (SLIC) under DC1 20 dB instance, the highest accuracy can be achieved at λ = 1e-3, β = 1e2 and λ_C=0.001. However, the highest accuracy under DC2 30 dB can be obtained in a different combination of λ = 1e-4, β = 1e-3 and λ_C = 1000.

(3) The AMUA algorithm could achieve relatively satisfactory results on test data sets by effectively determining regularization parameters adaptively. Figs. 6 and 7 demonstrate that the bubbles obtained by the AMUA share the same colour as the high accuracy bubbles obtained by MUA.

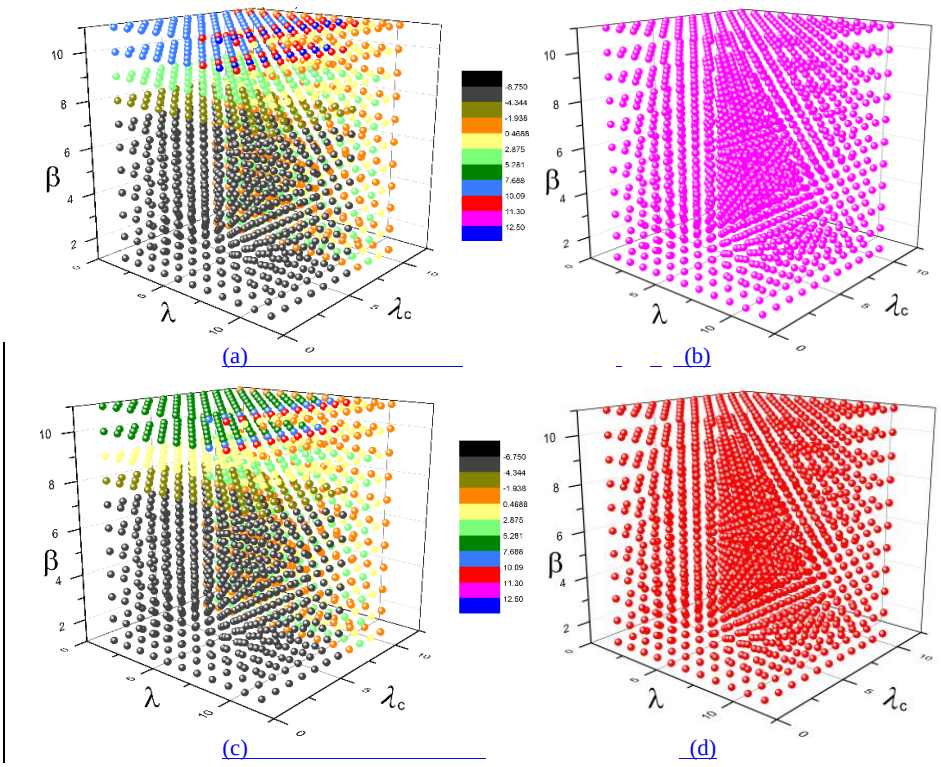


Fig. 6. SRE in relation with regularization parameters λ, β and λ_C for DC1 20dB

5.2. Sensitivity to the noise variance estimated from the HRSI

The noise variance δ_n^2 estimated following the method proposed in reference [43] is a crucial parameter in the ADMM and affects the conduct of the unmixing algorithms. However, the estimated results are usually unstable. Experiments are performed on the data cube DC2 (20 dB), DC2 (30 dB), DC2(40 dB) and the Cuprite to understand the effects effectively. The fluctuation of noise (FR) is defined as follows:

$$\mathrm{FR} = \frac{\mathrm{VNV}-\mathrm{TNV}}{\mathrm{TNV}}, \tag{14}$$

where TNV represents the true noise variance of the HRSI. VNV denotes the estimated noise variance for the HRSI. FR represents the fluctuation degree, which ranges within [-1, -0.01, -0.05, -0.1, -0.5, 0, 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 1000 and 10000]. Different VNV values could be obtained for sensitive experiments by using (15), providing the TNV and different FR values. Notably, FR = 0 means that the VNV is estimated accurately, which is equal to the truth value. Meanwhile, FR = -1 means the VNV = 0.

$$\mathrm{VNV} = \mathrm{TNV} + \mathrm{FR} \times \mathrm{TNV}\,. \tag{15}$$

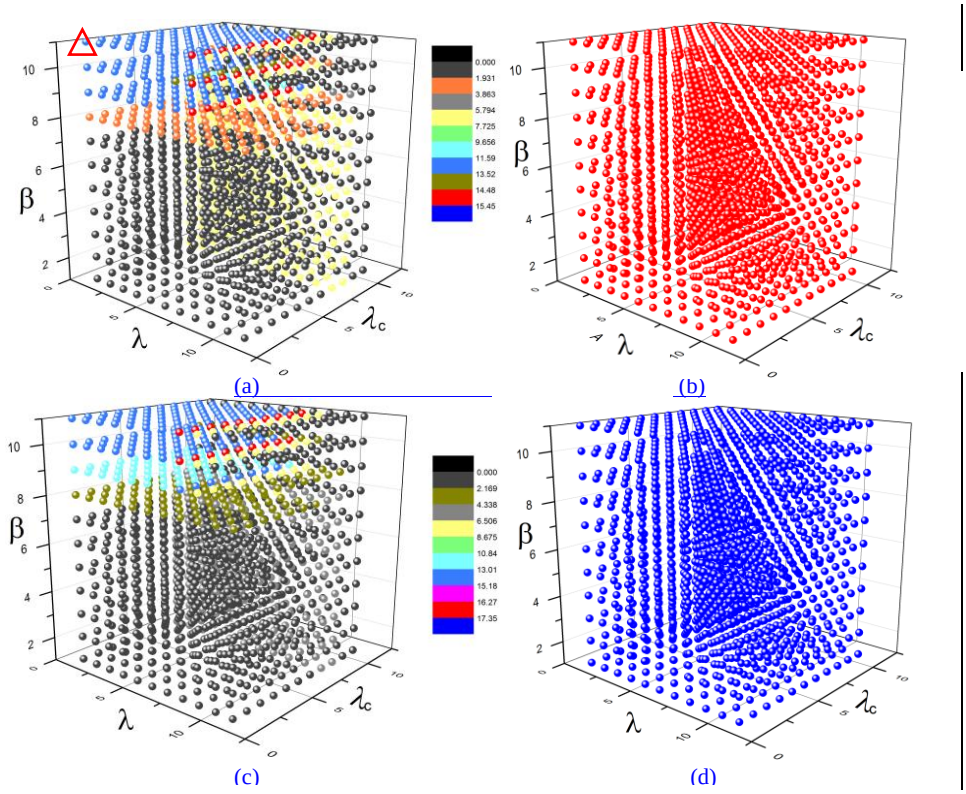


Fig. 7. SRE in relation with regularization parameters λ, β and λ_C for DC2 30dB

In the AMUA, the different values of the VNV are utilized to obtain different values for $\delta_n^{(t)}$ to derive various unmixing results. The relationship curves between SRE and the FR are shown in Fig. 8. The horizontal and vertical axis denote FR and SRE , respectively. The turquoise and yellow marker denote the SRE obtained with the true noise variance. The red and blue markers denote the SRE obtained with fluctuating estimated noise variance. From the Fig. 8, the following conclusions can be drawn.

1. In the absence of noise with FR = −1, which is impractical, the unmixing accuracy is low (i.e. the SRE falls to 1.667 in Fig. 8(b)).
2. The small fluctuation of the estimated noise variance cannot affect the unmixing performance significantly, such as in Fig. 8(b). In this figure, the SRE(SLIC) at location (FR = 0.01) is 17.595, whilst that obtained with the true noise is 17.6, demonstrating a difference of 0.005.
3. The unmixing accuracy decreases with the increase in the fluctuation of the VNV. For instance, in Fig. 8(a), the SRE(SLIC) at the location FR = 0 is 14.488, whilst that at FR = 1000 decreases to 14.327.

The ground truth for the Cuprite data is not provided. Thus, the true noise variance in HRSI is estimated by the method in [43], and the value(TNV) is 7.2659e-06, which is regarded as the truth noise variance. Different VNV values are employed in the AMUA to generate different abundance maps, as shown in Figs. 9,10 and 11. The abundance maps in each subfigure (f) are regarded as the ground truth maps. These maps reveal that the accuracy of AMAU is relatively insensitive to the fluctuation of the VNV. For example, in Fig. 9, most sub-figures are similar to the ground truth map, except for the maps in Figs. 9(a) and 9(h), which are far away with extreme FR values. The FR in Fig. 9 (a) is -1, which means that the noise variance VNV is 0, namely the HRSI is in the absence of noise. For FR in Fig. 9 (h) is 10000, which means that the fluctuation of the noise variance is extremely large.

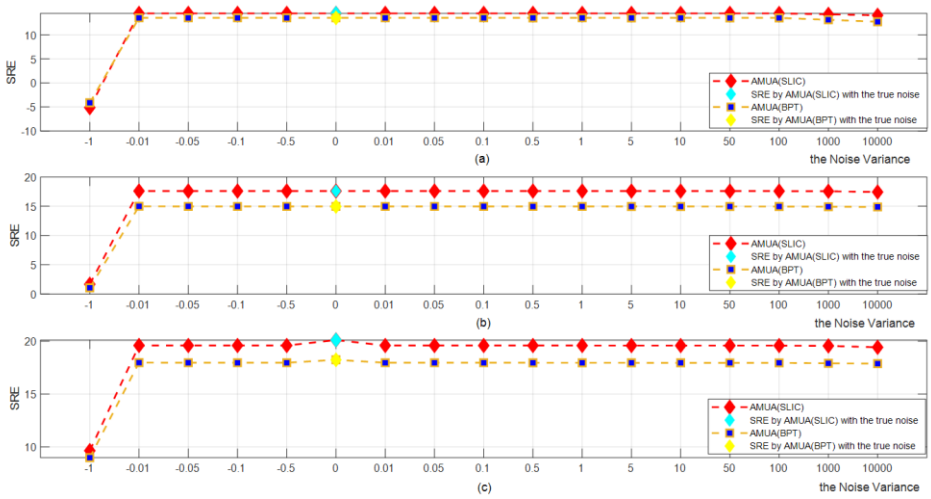


Fig. 8. SRE in relation to noise variance fluctuation for DC2.

The results generally illustrate the robustness of the AMUA to the small fluctuation of the VNV.

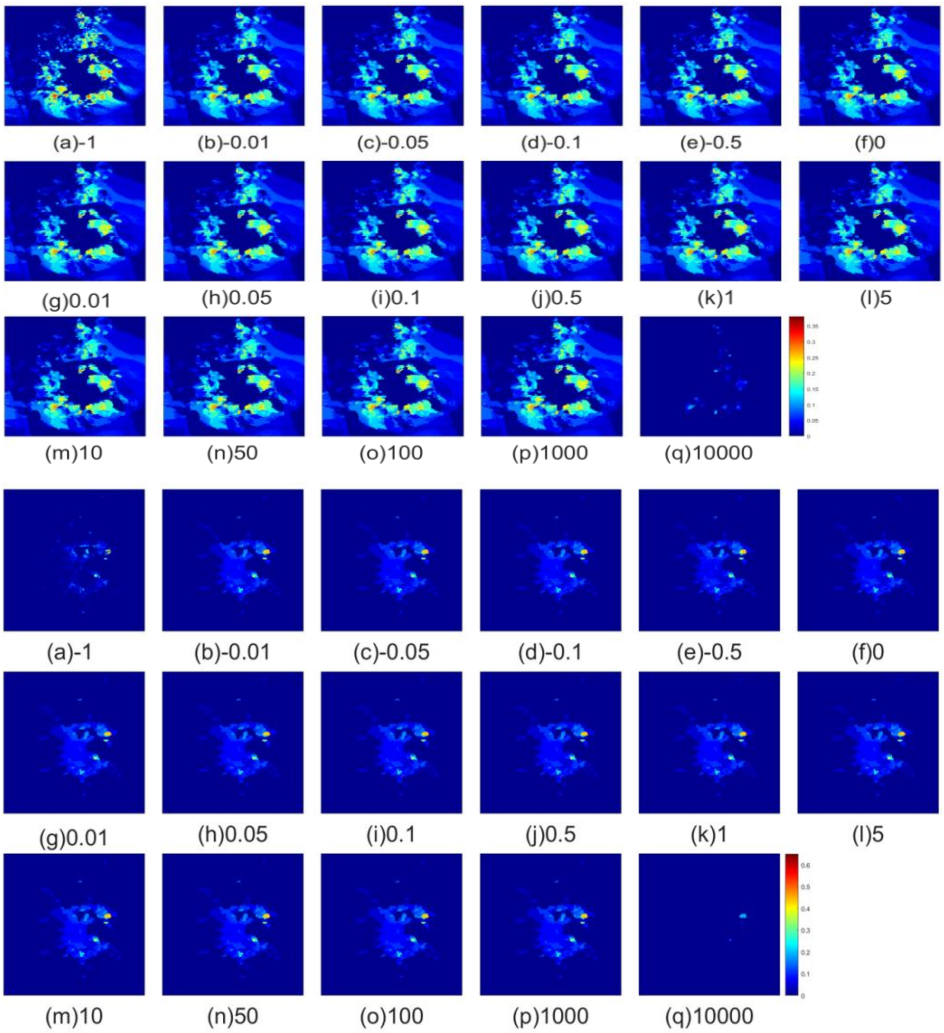


Fig. 10. The Buddingtonite abundance images considering the noise variance fluctuation

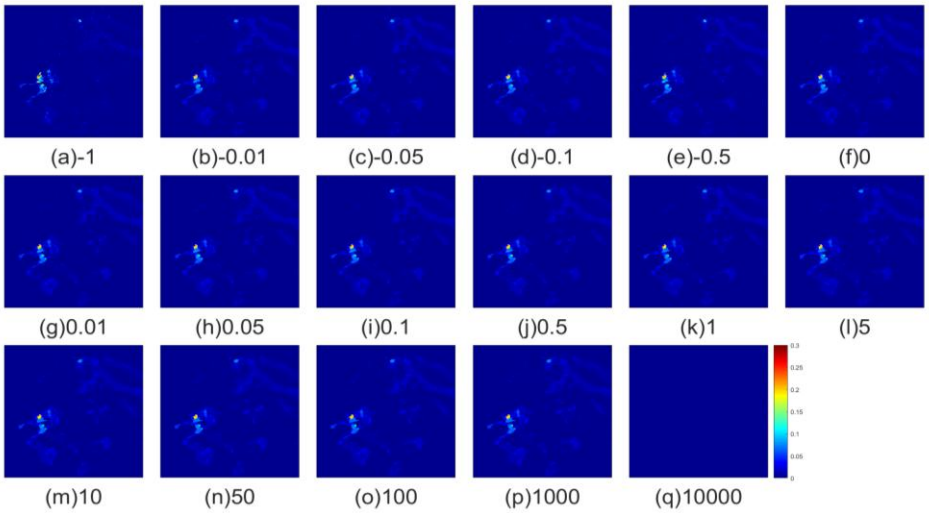


Fig. 11. The Chalcedony abundance images considering the noise variance fluctuation.

5.3. Sensitivity to the variance of the abundance updated at each iteration.

The detailed process of updating the regularization parameters in AMUA is presented as follows:

$$\begin{aligned}\lambda^{(t)} &= \sqrt{2}(\delta_n^{(t)})^2/\delta_X^{(t)}(\text{where } \delta_X^{(t)} = \text{std}(\widehat{\mathbf{X}}^{(t)}))\,,\\ \lambda_c^{(k)} &= \sqrt{2}(\delta_{Y_C}^{(k)})^2/\delta_{X_C}^{(k)}(\text{where } \delta_{X_C}^{(k)} = \text{std}(\mathbf{X}_c^{(k)}))\,,\\ \beta^{(t)} &= (\delta_n^{(t)})^2/\delta_\beta^{(t)};\ (\text{where } \delta_\beta^{(t)} = \text{std}(\widehat{\mathbf{X}}^{(t)} - \widehat{\mathbf{X}_D}))\,.\end{aligned} \tag{16}$$

The regularization parameters are updated according to $\widehat{\mathbf{X}}^{(t)}$, which is the estimated abundance obtained at iteration t.

For comparison, in the process of solving AMUA, $\widehat{\mathbf{X}}^{(t)}$ is replaced with the true abundance $\mathbf{X}$ denoted as AMUA-GT. That is, the parameters are set as follows:

$$\begin{aligned}\lambda^{(t)} &= \sqrt{2}(\delta_n^{(t)})^2/\delta_X^{(t)}(\text{where } \delta_X^{(t)} = \text{std}(\mathbf{X}))\,,\\ \lambda_c^{(k)} &= \sqrt{2}(\delta_{Y_C}^{(k)})^2/\delta_{X_C}^{(k)}(\text{where } \delta_{X_C}^{(k)} = \text{std}(\mathbf{X}_c^{(k)}))\,,\\ \beta^{(t)} &= (\delta_n^{(t)})^2/\delta_\beta^{(t)};\ (\text{where } \delta_\beta^{(t)} = \text{std}(\mathbf{X} - \widehat{\mathbf{X}_D}))\,.\end{aligned} \tag{17}$$

The abundance images estimated by AMUA, AMUA-GT and other different algorithms are shown in Fig. 12 and Fig. 13 ,with subfigure (a) under 20 dB, subfigure (b) under 30 dB and subfigure (c) under 40 dB. The corresponding SRE values are also listed in Table 5.

From Fig. 12 and Fig. 13, It can be revealed that the difference between the abundances estimated by AMUA-GT and AMUA is insignificant. And the same conclusion can be drawn in Table 5. For DC1 with 30 dB instance, the SRE values of AMUA-GT (BPT) and AMUA(BPT) are 18.281 and 18.217, respectively. The difference is only 0.052. For DC1 with 40 dB instance, the SRE values of AMUA-GT

(BPT) and AMUA(BPT) are 21.959 and 21.806, respectively. The tiny difference is 0.153. For DC2 with 20 dB instance, the SRE values of AMUA-GT (SLIC) and AMUA (SLIC) are 14.77 and 14.488, respectively. The difference 0.286 is also very small. So the AMUA is insensitive to the variation of the abundance updated at each iteration.

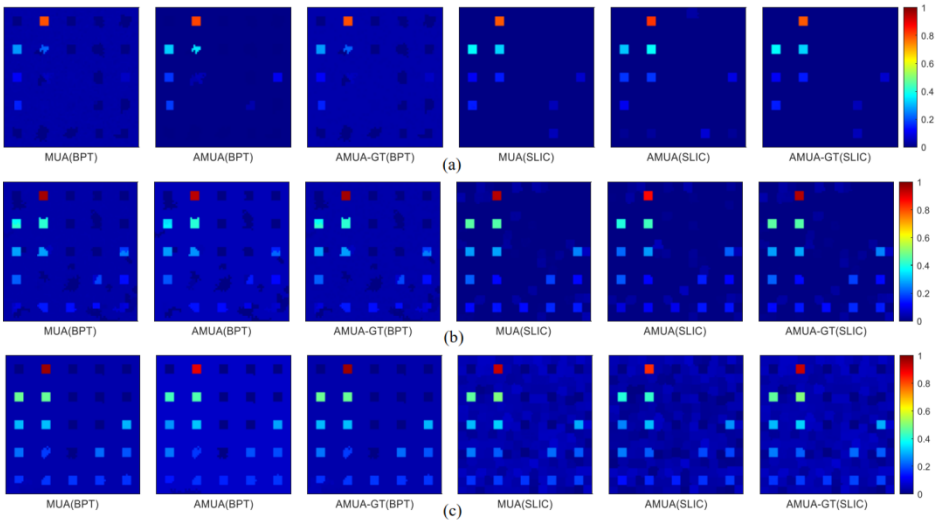


Fig. 12. The first comparison of abundance images estimated by different algorithms

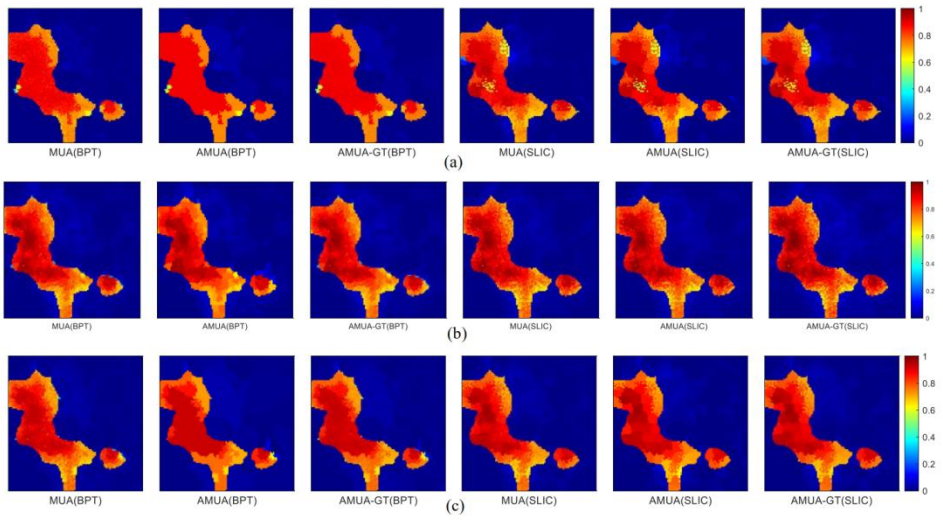


Fig. 13. The second comparison of abundance images estimated by different algorithms

Table 5. The comparison of SRE values by different algorithms.

Algorithms / Data	DC1 data cube			DC2 data cube		
	20 dB	**30 dB**	**40 dB**	**20 dB**	**30 dB**	**40 dB**
MUA(BPT)	13.393	18.257	21.8	13.928	16.963	18.54
AMUA-GT(BPT)	13.435	18.281	21.959	13.788	16.134	18.68
AMUA(BPT)	12.715	18.217	21.806	13.554	14.9704	16.32
MUA(SLIC)	11.346	15.731	22.126	14.757	18.328	20.52
AMUA-GT(SLIC)	11.380	15.770	22.302	14.774	18.184	22.07
AMUA(SLIC)	10.530	13.832	20.17	14.488	17.60	20.38

6. Conclusion

The efficient AMUA algorithm is proposed in this paper for unmixing hyperspectral images, which can be applied to national land resources detection. The AMUA model is constructed by applying the MAP to the MUA model. The relationships among regularisation parameters, estimated abundances and estimated noise variances are explored and applied to ADDM to update the regularisation parameters automatically during the optimization process.

Two simulated hyperspectral datasets and one real HRSI were employed to verify the effectiveness and accuracy of the proposed algorithm. All the observations prove that the calculation efficiency is markedly improved without manual parameter selection at the cost of negligible accuracy. A series of sensitivity experiments were also undertaken to demonstrate the robustness of the AMUA algorithm.

The limitation lies in the fact that the noise in the observed HRSI is assumed as independent and identically Gaussian distribution, which is not suitable for many practical situations. Therefore, the sparse unmixing for HRSI contaminated with non-Gaussian noise will be studied in the future.

References

1. Zhang T., Hou M., Zhou T., Liu Z., Cheng W., and Cheng Y.,“Land-use classification via ensemble dropout information discriminative extreme learning machine based on deep convolution feature,”Computer Science and Information Systems., vol. 17, no.2, pp.427-443,2020.
2. Liu H., Li C., and Xu L.,“Dimension Reduction and Classification of Hyperspectral Images based on Neural Network Sensitivity Analysis and Multi-instance Learning,”Computer Science and Information Systems . , vol. 16, no.2, pp.443-468,2019.
3. Cheng, R., Yu, W., Song, Y., Chen, D., Ma, X. and Cheng, Y. Intelligent Safe Driving Methods Based on Hybrid Automata and Ensemble CART Algorithms for Multihigh-Speed Trains[J]. IEEE Transactions on Cybernetics,vol. 49, no.10, pp. 3816-3826,2019.
4. Liu, G., Chen, X., Zhou, R., Xu, S., Chen, Y.C. and Chen, G. Social learning discrete Particle Swarm Optimization based two-stage X-routing for IC design under Intelligent Edge Computing architecture[J]. Applied Soft Computing. 10, 10721,2021
5. Liu, G., Zhang, X., Guo, W., Huang, X., Liu, W., Chao, K. and Wang, T. Timing-Aware Layer Assignment for Advanced Process Technologies Considering Via Pillars. IEEE

Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no.6, pp. 1957-1970,2022.
6. Cheng, H., Wu, L., Li, R., Huang, F., Tu, C. and Yu, Z. Data recovery in wireless sensor networks based on attribute correlation and extremely randomized trees [J]. Journal of Ambient Intelligence and Humanized Computing, vol. 12, no.1, pp245-259,2021.
7. Liu, G., Zhu, Y., Xu, S., Chen, Y.C. and Tang H. PSO-Based Power-Driven X-Routing Algorithm in Semiconductor Design for Predictive Intelligence of IoT Applications. Applied Soft Computing, 114: 108-114,2022.
8. Li F. "Low-Rank and Spectral-Spatial Sparse Unmixing for Hyperspectral Remote Sensing Imagery," Wireless Communications and Mobile Computing., vol. 2021, no.1, pp.1-14, 2021.
9. Yu, Z., Zheng, X., Huang, F., Guo, W., Sun, L. and Yu, Z. A framework based on sparse representation model for time series prediction in smart city[J]. Frontiers of Computer Science, 15(1): 1-13,2021.
10. Lu, Z., Liu, G. and Wang, S. Sparse neighbor constrained co-clustering via category consistency learning [J]. Knowledge-Based Systems, 201, 105987,2020.
11. Shen, S., Yang, Y. and Liu, X. Toward data privacy preservation with ciphertext update and key rotation for IoT [J]. Concurrency and Computation: Practice and Experience, e6729. https://doi.org/10.1002/cpe.6729,2021.
12. Wang, Le et al., "A survey of methods incorporating spatial information in image classification and spectral unmixing," International Journal of Remote Sensing., vol. 37, no.16, pp.3870-3910, 2016.
13. Feng, Ruyi et al., "Rolling guidance based scale-aware spatial sparse unmixing for hyperspectral remote sensing imagery," Remote Sensing., vol.9, no. 12 , pp.1218-1221, 2017.
14. Shi, Zhenwei et al., "Collaborative Sparse Hyperspectral Unmixing Using $ l_ {0} $ Norm," IEEE Transactions on Geoscience and Remote Sensing., vol. 56, no. 9, pp.5495-5508, 2018.
15. Ertürk, Alp, Marian-Daniel Iordache, and Antonio Plaza, "Sparse unmixing-based change detection for multitemporal hyperspectral images," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing., vol. 9, no.2, pp.708-719, 2015
16. Qi, Lin et al., "Region-Based Multiview Sparse Hyperspectral Unmixing Incorporating Spectral Library," IEEE Geoscience and Remote Sensing Letters., vol. 7, no.16, pp.1140-1144, 2019.
17. Palsson, Burkni et al., "Hyperspectral unmixing using a neural network autoencoder," IEEE Access., vol.6, pp.25646-25656, 2018.
18. Dai, Y., Wang, S., Chen, X., Xu, C. and Guo, W..Generative adversarial networks based on Wasserstein distance for knowledge graph embeddings[J]. Knowledge-Based Systems, 190: 105165,2020.
19. Li, X. Y., Lin, W., Liu, X., Lin, C., Pai, K. and Chang, J. Completely Independent Spanning Trees on BCCC Data Center Networks with an Application to Fault-Tolerant Routing[J]. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, vol.33, no.8 pp. 1939-1952,2022.
20. Iordache, Marian-Daniel, José M. Bioucas-Dias, and Antonio Plaza, "Sparse unmixing of hyperspectral data," IEEE Transactions on Geoscience and Remote Sensing., vol.49, no.6, pp.2014-2039,2011.
21. Iordache, Marian-Daniel, José M. Bioucas-Dias, and Antonio Plaza."Total variation spatial regularization for sparse hyperspectral unmixing," IEEE Transactions on Geoscience and Remote Sensing., vol. 50, no.11, pp.4484-4502, 2012.
22. Zhang, Shaoquan et al. "Spectral–spatial weighted sparse regression for hyperspectral image unmixing," IEEE Transactions on Geoscience and Remote Sensing., vol.56, no.6, pp.3265-3276,2018.

23. Borsoi, Ricardo Augusto et al. “A Fast Multiscale Spatial Regularization for Sparse Hyperspectral Unmixing,” IEEE Geoscience and Remote Sensing Letters., vol.4, no.16, pp.598-602,2018.
24. Cheng, Y., Jiang, H., Wang, F., Hua, Y., Feng, D., Guo, W. and Wu, Y. Using High-Bandwidth Networks Efficiently for Fast Graph Computation [J]. IEEE Transactions on Parallel and Distributed Systems, vol.30, no.5, pp. 1170-1183,2019.
25. Wang, S., Wang, Z., Lim, K. L., Xiao, G. and Guo, W. Seeded random walk for multi-view semi-supervised classification [J]. Knowledge-Based Systems, 222:107016, 2021.
26. Zhang, H., Li, J. L., Liu, X. M. and Chen, D. Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection [J]. Future Generation Computer Systems, 122: 130-143,2021.
27. Zhang, Y., Lu, Z. and Wang, S. Unsupervised feature selection via transformed auto-encoder[J]. Knowledge-Based Systems, 215: 106748,2021 .
28. Fu, Y. G., Ye, J. F., Yin, Z. F., Chen, L., Wang, Y. and Liu, G. Construction of EBRB classifier for imbalanced data based on Fuzzy C-Means clustering[J]. Knowledge-Based Systems, 234: 107590,2021.
29. Arridge, Simon et al. “Solving inverse problems using data-driven models,” Acta Numerica., vol. 28, pp. 1-174, 2019.
30. Ren, Dongwei et al. “Simultaneous Fidelity and Regularization Learning for Image Restoration”., arXiv preprint arXiv:1804.04522 , 2018.
31. Cao, Chunhong et al. “Hyperspectral Image Denoising via Subspace-Based Nonlocal Low-Rank and Sparse Factorization,” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing., vol.3, no.12, pp. 973-988,2019.
32. Fu, Y. G., Zhuang, J. H., Chen, Y. P., Guo, L. and Wang, Y. A framework for optimizing extended belief rule base systems with improved Ball trees [J]. Knowledge-Based Systems, 210: 106484,2020.
33. Liu, G., Chen, Z., Zhuang, Z., Guo, W. and Chen, G. A unified algorithm based on HTS and self-adapting PSO for the construction of octagonal and rectilinear SMT[J]. Soft Computing, vol.24, no.6, pp.3943-3961,2020.
34. Liu, N., Pan, J. Sun, C., Ch, Su. An efficient surrogate-assisted quasi-affine transformation evolutionary algorithm for expensive optimization problems [J]. Knowledge-Based Systems, 209: 106418,2020.
35. Huang, Zhen et al. “Maximum a posteriori adaptation of network parameters in deep models,” Conference of the International Speech Communication Association, ISCA, September 6-10, 2015, Dresden, Germany, pp. 1076-1080,2015.
36. Pereyra, M. . "Revisiting Maximum-A-Posteriori Estimation in Log-Concave Models." SIAM Journal on Imaging Sciences., vol. 12, no.1, pp.650-670,2019.
37. Feng, Ruyi, Yanfei Zhong, and Liangpei Zhang. “Adaptive spatial regularization sparse unmixing strategy based on joint MAP for hyperspectral remote sensing imagery,” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing., vol.9, no.12, pp. 5791-5805,2016.
38. Chen, N. N., Gong, X. T., Wang, Y. M., Zhang, C. Y. and Fu, Y. G. Random clustering forest for extended belief rule-based system [J]. Soft Computing, vol.25, no.6, pp.4609-4619,2021.
39. Fu, Y. G., Huang, H. Y., Guan. Y., Wang, Y., Liu, W. and Fang, W. EBRB cascade classifier for imbalanced data via rule weight updating [J]. Knowledge-Based Systems, 223: 107010,2021..
40. Guo, L., Li, M. and Xu, D. Efficient Approximation Algorithms for Maximum Coverage with Group Budget Constraints [J]. Theoretical Computer Science, 788:53-65,2019.
41. Li H. , Lin Z. . “Accelerated Alternating Direction Method of Multipliers: an Optimal O(1/K) Nonergodic Analysis”, Journal of Scientific Computing, vol.79, no.2, pp. 671-699,2019.

42. Babacan, S. Derin, Rafael Molina, and Aggelos K. Katsaggelos. “Bayesian compressive sensing using Laplace priors,” IEEE Transactions on image processing., vol.19, no.1, pp.53-63,2009.
43. Bioucas-Dias J M, Nascimento J M P. “Hyperspectral subspace identification,” IEEE Transactions on Geoscience and Remote Sensing. vol. 46, no.8, pp.2435-2445,2008.
44. Miguel A et al. “Hyperspectral image segmentation using a new spectral unmixing-based binary partition tree representation,” IEEE Transactions on Image Processing., vol. 23, no.8, pp.3574-3589,2014.
45. R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, “SLIC superpixels compared to state-of-the-art superpixel methods,” IEEE transactions on pattern analysis and machine intelligence. vol. 34, no.11, pp.2274–2282,2012.

Yalan Li received the B.S. degree from Hunan University of Science and Technology, Xiangtan, China in 2004 and received the M.S., and Ph.D. degrees in communication engineering from Huazhong Normal University, Wuhan, China, in 2007, and 2017, respectively. He is currently an Associate Professor with the School of Physics and Electronic Electrical Engineering, XiangNan University, Chenzhou, China. His research interests include image processing and pattern recognition, etc.

Qian Du received the B.S. degree from Qufu Normal University, Qufu, China in 2004 and received the B.S., M.S. degrees in communication engineering from Huazhong Normal University, Wuhan, China, in 2004. She is currently an Associate Professor with the School of Information Science and Engineering, Linyi University, Linyi, China. Her research interests include wireless networks algorithm and Internet of Things, etc.

Yixuan Li received the B.S. degree in Nankai University Binhai College, Tianjin, China, in 2019. He is currently pursuing the master’s degree with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang, China. His research interests include UAV-aided communications, intelligent reflecting surface and wireless power communication networks.

Wenwu Xie received the B.S., M.S., and Ph.D. degrees in communication engineering from Huazhong Normal University, Wuhan, China, in 2004, 2007, and 2017, respectively. He is currently an Associate Professor with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang, China. His research interests include communication algorithm, such as channel estimation, equalizer, encoding/decoding, etc.

Jing Yuan received the Ph.D. degree in Information and Communication Engineering in 2019 from Tsinghua University, Beijing, China. She is currently an Associate Professor in the Institute of Disaster Prevention. Her current research interests include Machine Learning, Hyperspectral Image Unmixing and Sparse Representation.

Shang Lin Li received his doctor degree in Computer Science and Technology from Computer and Information Department of Hefei University of Technology in 2017. He is currently an associate professor of Computer Application Technology in the

Computer Science and Artificial Intelligence department at Xiangnan University. His research interests include Computer Graphic, Rapid Modeling and Machine Learning.

Qi Chen received the B.S. degree in control technology and instruments from China Jiliang University, Hangzhou,China,in 2002 and received the M.S. and Ph.D. degrees in communication engineering from Huazhong Normal University, Wuhan, China, in 2006, and 2019, respectively. She is also an Associate Professor at the College of Computer and Information Engineering of Hubei Normal University, Huangshi, China.Her research interets include image processing and computer vision.

Received: August 28, 2022, Accepted: December 15, 2022.

Computer Science and Information Systems 20(2):573–593 https://doi.org/10.2298/CSIS220715010L

Evaluation of Smart City Construction and Optimization of City Brand Model under Neural Networks

Yingji Li [1], Yufeng Qian [2], Qiang Li [3], and Linna Li [4*]

[1] School of Humanities and Management, Yunnan University of Chinese Medicine, 650500, Kunming, China
lyj.123@163.com
[2] School of Science, Hubei University of Technology, Wuhan, 430068, China
yfqian@whu.edu.cn
[3] School of Economics and Management, Shanghai Technical Institute of Electronics & Information, Shanghai 201411, China
339865742@qq.com
[4] School of Design, The University of Melbourne, Melbourne,3052, Australia
linnal@student.unimelb.edu.au

Abstract. The study aims to avoid the phenomenon that thousand cities seem the same in the construction of smart cities and the efforts of all walks of life are jointed to implement the construction of smart cities and the creation of city brands. First, the basic theory of smart city construction is introduced. Second, the restricting and promoting factors influencing smart city construction and development are analyzes, and the evaluation system of smart city development is established. Then, the model for smart city construction and development based on the neural network is implemented. Finally, some domestic cities are selected as the dataset to build a training model, and the city brand optimization strategy is proposed. On this basis, an evaluation system for smart city development based on the intelligent neural network and Grey relational analysis Back Propagation Neural Network (GRA-BPNN) is obtained. The entropy weight method (EWM), grey relational analysis (GRA) method and the evaluation method based on Technique for Order Preference by Similarity to an Ideal Solutionv (TOPSIS) are regarded as the members of the control group, and the results of different methods for evaluating the development of smart cities are compared. The results show that the modeling of smart city construction based on neural networks can help to implement an evaluation model for smart city construction and development, and it can help evaluate smart city construction and development accurately. And then the corresponding strategies are proposed to speed up the construction of smart cities with its local characteristics, and then city brands are built. Compared with other evaluation algorithms, the performance of the algorithm proposed is better and more stable and the evaluation results are more reasonable than those of the others, which can prove that the evaluation algorithm is feasible and scientific. This study provides a new idea for the application of deep learning to smart city construction and city brand building.

Keywords: Neural network model; Smart city; Construction and development; City brand.

* Corresponding author

1. Introduction

With the progress of urbanization, the demand for economic development, resource allocation, and sustainable development of cities becomes higher. At the end of 2019, urban population reaches 850 million, and the urbanization rate is over 60%. With the increase of the urban population, the need for a higher level of urban management is becoming more and more urgent. Therefore, the concept of constructing a smart city and a new satellite city in large cities is proposed to solve the problem caused by urbanization, and the traditional urban management model needs to be improved [1, 2].

There are following studies on the role of smart cities. The construction of smart cities or digital cities has a positive effect on improving the traffic conditions in big cities, increasing the utilization rate of traffic facilities, speeding up the development mode of green cities, promoting scientific governance, changing the old production mode of the city, improving the living standard of residents and making the residents feel happy [3, 4]. With the accelerating of urbanization, urban development can drive regional development in the brand-led competition in the international market. The Inter-regional competition has become the competition of comprehensive strength with cities as the core, the carrier and the platform. Generally speaking, smart cities can improve the happiness of residents and increase the competitiveness of cities. Research on the theoretical framework and evaluation methods of smart cities: as an important element in urban competition, city brand building gradually becomes a major approach to the development of many cities, and city brand building will inevitably become an important part of urban development. At present, there is no universally acknowledged specific expression of the connotation of urban brand. Different researchers have put forward their own theoretical framework and evaluation systems from different aspects, and have accordingly selected different research methods. Xiang et al. (2019) pointed out that some smart cities have been constructed, but the relevant evaluation system is not established. Therefore, according to the concept of smart city construction, the state of the current smart city construction is obtained by modeling and calculation [5]. And the evaluation index system is constructed. The closest correlation degree of the six indexes is discussed, and the key indexes affecting smart city development are analyzed. Regarding the development status and existing problems of smart cities, Su et al. (2019) argued that smart city construction is an important direction of global urban development and reflects the people's wish for life [6]. The construction of smart city is also a long-term task to promote urbanization and urban development. The new urbanization takes wisdom, and green ecology as the core, and more and more cities plan to construct a city with green ecology, wisdom and wisdom urbanization. Landscape pattern optimization is one of the important ways to realize regional ecological security. But with the rapid development of urbanization, ecological space has been compressed. How to coordinate the contradiction between regional ecological security and socio-economic development has become the main problem of land use. Ming and Wang (2019) also pointed out that with the growth of urban population, the rapid development of smart cities has become the focus of urban regional development [7]. They said that smart healthcare is an indispensable part of smart city construction. However, the security of data and the timeliness of services are the problems faced by the current intelligent medical system. The concept and function of neural network are studied as follows. Back Propagation Neural Network (BPNN) is a multilayer feedforward neural network developed on the basis of the sensor network, and it is

widely used in the Internet field. The classification of BPNN is complex, and BPNN has excellent multidimensional function mapping ability. BPNN includes input layers, hidden layers and output layers. In essence, the objective function of the algorithm based on BPNN uses the network error square and the gradient descent method to calculate the minimum function. Nowadays, neural network technology is widely used in all walks of life,attracting wide attention from many experts and scholars. Vekinis and Constantoudis (2020) made predictions by using multi-layer neural networks, including surface roughness and the geometric shape of a given atomic force microscope (AFM) [8]. Dubourg et al. (2019) simulated data by training AFM, and the measurements show a good fitting between the training set and the test set. Machine learning is used to evaluate the samples and fast and efficient selection is realized, reducing the operation cost of AFM measurement [9]. Ostad-Ali-Askari and Shayannejad (2021) used artificial neural networks to calculate the drainage spacing under unstable conditions in areas with different soil properties and drainage spacing. The network designed contains a hidden layer with four neurons, and the drainage spacing calculated by the method is consistent with the actual value, and has higher accuracy than other methods [10] . In addition, machine learning is also widely used in different fields. For example, in the analysis of residential land prices in smart cities, four nonlinear machine learning algorithms are used to build a block residential land price prediction model, which provides a theoretical basis for in-depth understanding of urban residential land prices. In the smart city, the integration of AI and the IoT technology is applied to the smart home to realize the construction of the whole house smart home system. In the smart city fire safety, the improved naive Bayes algorithm is used to improve the sensitivity of fire equipment identification. "Machine learning" can be used in urban drainage design to predict hazards.

It is found that neural network technology is mainly used in the field of natural science, and it is less used to research social life and business activities. In order to solve the problem of "multiple cities look the same" in the construction of smart cities that appeared in many places, people should effectively implement the construction of smart cities and the building of city brands. Therefore, the innovation of this study is that neural network technology is used to study the optimization of smart city construction and city brand building, and a new computer technology is proposed and used in the evaluation of city brand construction. The technology provides a new perspective for the evaluation of smart city construction and development. In the information society, a variety of new computer technologies emerge endlessly, and a technology that can be used to evaluate smart city construction is born. Here, the relevant theories of urban brand optimization are analyzed and studied, and the factors affecting the construction and development of smart cities are analyzed. Then, BPNN is introduced to establish the evaluation model of the construction and development of smart cities, and the urban brand optimization strategy is proposed to evaluate the construction and development of smart cities and optimize the city brand model. The establishment of a BPNN model simplifies the manual calculation, and calculates the correlation among the indicators of the smart city through the grey relational theory to ensure the validity of the assessment of smart city construction. The data output by the BPNN algorithm based on grayscale correlation can be more accurate. This study solves the problem of evaluating the development potential of smart cities, provides a reliable data model for the future development and construction of smart cities, and plays a

positive role in improving the development of smart cities and ensuring the ability to build smart cities.

2. Theories of Smart City Construction and Optimization of the City Brand Model

2.1. BPNN

Neural networks. The biological neural network is used to construct the structure of the model based on the artificial neural network (ANN), which is a combination of many artificial neurons. ANN can adjust its internal structure with the change of the external environment. It can also be used as a modeling tool to characterize and model the functional relationship between the input and the output, or to analyze the relations between data. The content of the neural network is an algorithm constructed by the logic of the external environment or human society. The invention of the neural network is inspired by the neural network operation principle of natural biology (human or mammals) [11,12]. ANN is an active learning model, which optimizes the model according to the statistical results. Its cavities are calculated by statistical laws, and many mathematical descriptions in local structural space are obtained. Also, mathematical statistics are mainly used for learning and calculation in the field of machine perception, as well as for simulating the artificial perception system. Finally, ANN obtains the most basic decision and judgment ability. Compared with the traditional logic operation model, this self-learning model is promising. Chen (2020) studied the application of deep learning to smart city construction [13].

ANN does not need to directly reveal the functional relationship between the input, the output, and the mathematical equation. It only needs to use its network independently to learn the laws and the rules between data. When the input value is given, the corresponding output value is obtained.

BPNN. The BPNN system is an artificial neuron model with a multi-layer network. Its basic idea is that the gradient descent method is used to minimize the mean square error between the actual output value given by the model and the expected through gradient search. Figure 1. shows the topological structure of the BPNN model [14, 15].

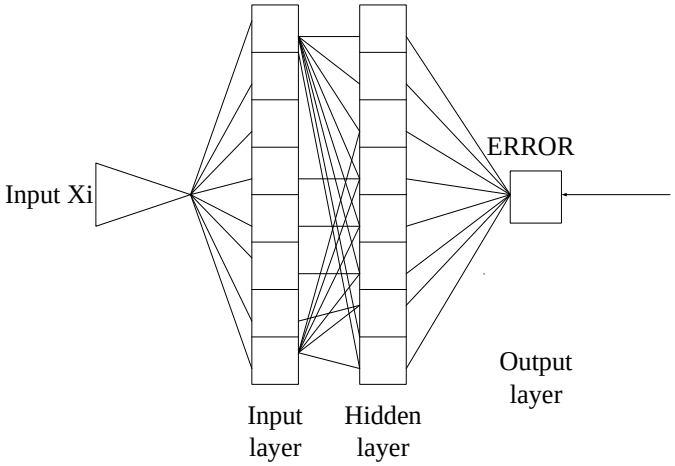


Fig. 1. Topological structure of the BPNN model

The activity of biological neurons is presented by a simple mathematical model. It is used to receive the signals transmitted by other multiple neurons, which are aggregated into total signals. The total signals are compared with the thresholds. If the threshold is greater than the limited, the excitation signal is output. If the threshold is less than the limited, the excitation signal is in the inhibitory state. A plane is divided into two parts by a straight line, and the three-dimensional space is divided into two parts by the delayed plane. A dimension hyperplane n-1 divides the n -dimension space into two parts, which belong to different categories. Such classifiers are called neurons. The equation used in dividing the plane is as follows:

$$ax + by + c = 0 \,. \tag{1}$$

The left side of the equation can be greater than zero or less than zero, which respectively represents the point on one side or the other side of the line. This equation is extended into the dimensional space. The high-dimensional form of the line is called a hyperplane. Its equation is as follows:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + a_0 = 0 \,. \tag{2}$$

The neuron in the model outputs 1 when h is greater than 0 or less than 0. It is used to divide the feature space in half and the two lobes belong to two categories, as shown in Figure 2.

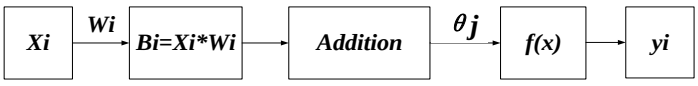


Fig. 2. M-P neuron model

After it is input and learned, the direction of the feature vector is changed. The absolute value of the weight is the influence of the input signal on neurons.

The activation function has six categories:

a. Oblique function: when the absolute value of x is less than c c, f(x)=kx+c. When x is greater than c, it is equal to T. When x is less than -c, it is equal to -T.

b. Threshold function: when x is greater than c, f(x)+1; when x is less than -c, it is equal to 0.

c. The expression of the nonlinear function is shown in equation (3):

$$f(x) = \frac{1}{1+e^{-\alpha x}} (x \in R) . \tag{3}$$

d. The derivative of the Sigmoid function is as follows:

$$f'(x) = \frac{\alpha e^{-\alpha x}}{(1+e^{-\alpha x})^2} = \alpha f(x)[1 - f(x)] . \tag{4}$$

e. The bipolar Sigmoid function is as follows:

$$h(x) = \frac{2}{1+e^{-\alpha x}} - 1(x \in R) . \tag{5}$$

f. The derivative of the bipolar Sigmoid function is as follows:

$$h'(x) = \frac{2\alpha e^{-\alpha x}}{(1+e^{-\alpha x})^2} = \alpha \frac{1-h(x)^2}{2} . \tag{6}$$

α represents the weight of each node, and the results of the neural network need to be normalized to 0 – 1. When the transmission function is between 0-1, the difference between the results is significant. When the transmission function is more than 1, its variation will be significantly reduced (derivative or slope is relatively small), which has adverse effects on the performance of the BPNN algorithm. The BPNN algorithm uses the gradient information of the transmission function between each node during the process. If the input value is too high, it will lead to a small gradient value, causing the difficulty of adjusting the relationship between the weight and the threshold.

2.2. Grey relational theory

Grey relational analysis (GRA). GRA is a multi-factor analysis method that is often used in the grey system. For the correlation between the two systems, the degree of the correlation varies with time. In the development of the system, the degree of synchronous change is high, that is, the degree of the correlation between the two is high if the changing trends of the two are consistent. Therefore, the grey relational analysis method is based on the similarity or difference of the development trend between the two factors, that is, "grey relational degree", a method of measuring the relational degree between factors, and it is generally used to analyze the influence and contribution of each factor on the main body under different situations. The grey relational method can calculate the development trend of the curve according to its state and similarity and determine the consistency between districts and counties [16-18].

Grey relational algorithm. The flow chart of the grey relational algorithm is shown in Table 1 [19].

Table 1. Flow chart of the grey relational algorithm

Number	Steps
a	Indexes are set as follows: in $\mu_{ik}(i = 1,2,3,\ldots,n; k = 1,2,3,\ldots m)$, i represents the number of attributes, ω_k is the weight of the table is the weight value of k indicators, C_k is the characteristic proportion of the weight, and f_{ik} is the weight. The relationship of these factors is shown in equations 7–9. $f_{ik} = \frac{\mu_{ik}}{\sum_{i=1}^{n} \mu_{ik}}$ (7) C_k=-$\frac{1}{ln\, n} \sum_{i=1}^{n} f_{ik}\, ln\, f_{ik}$ (8) $\omega_k = \frac{1-C_k}{m-\sum_{i=1}^{m} C_k}$ (9)
b	If there are A_i objects, m indexes, and $Ai = (x_1,\ldots x_m), (i = 1,2,3,\ldots,n)$, and the best scheme is marked as $Z^+ = (max\, Z_{i1},\ldots, max\, Z_{im})$, the worst scheme is marked as$Z^- = (min\, Z_{i1},\ldots, min\, Z_{im})$, and the weight of m indexes is $\omega = \omega_1,\ldots,\omega_m$.
c	The grey relational coefficient is calculated as follows: $\xi(k) = \frac{min_i\{\, min_k\, \Delta_i\,(k)\} + \rho\, min_i\{\, min_k\, \Delta_i\,(k)\}}{\Delta_i(k) + \rho\, min_i\, min_k\, \Delta_i\,(k)}$ $= \frac{min_i\{\, min_k |x_0(k)-x_i(k)|\}+\rho\, min_i\{\, min_k |x_0(k)-x_i(k)|\}}{|x_0(k)-x_i(k)|+\rho\, min_i\{\, min_k |x_0(k)-x_i(k)|\}}$ (10) ρrepresents the resolution coefficient, and its range is between 0 and 1.
d	The grey relational degree r_i is calculated, and the equation is as follows: $r_i = \frac{1}{n} \sum_{i=1}^{n} \zeta_i(k)$ (11) n represents the number of sequence objects. The larger the value of r_i is, the greater the grey relational degree is. $\zeta_i(k)$ is the grey relational coefficient. When the value is greater than or equal to one-half of the value, the relational effect is obvious.
e	If the optimal sequence r_i is maximum, the maximum relational sequence is obtained.

2.3. Evaluation of smart city construction

Smart cities. A smart city is the advanced form of urban informatization after a digital city and it is constructed by the deep integration of informatization, industrialization, and urbanization. At present, smart city construction promotes the development of connotative urbanization, develops emerging industries, increases new economic growth points, upgrades traditional industries, social energy conservation, and emission reduction, and the transformation of economic development mode, and seizes the opportunity of the industrial revolution in the international competition.

In 2010, smart city construction is first proposed by a company in the United States. The company divides the city and sets up multiple systems with different functions, including the municipal system, the traffic system, and the system to expand its communication capacity, and the systems for water supply and energy supply. These systems do not work alone but cooperate mutually. Under this circumstance, smart city construction and development gradually become the focus of the research. Although the government has not issued specific documents to support the action, researchers in the academic circle have reached a consensus on this issue. Their ideas for smart city construction are that a smart city should have following characteristics: (1) the city management is smarter; (2) living in the city is more convenient; (3) public information is transparent; (4) the utility of resources is reasonable; (5) the ecological environment is livable; (6) the city's culture is inclusive of all kinds of non-mainstream ideas and lifestyles; (7) the interpersonal relationship of the residents is sincere and public services are convenient; (8) the city management is refined; (9) the living environment is livable; (10) the infrastructure is intelligent; (11) the network is safe [20, 21].

The proposal of smart city construction is based on cities' development. Computer technology is one of the emerging technologies, such as blockchain technology [22], cloud computing technology [23], big data technology [24], IoT (Internet of Things) [25], and the mobile wireless network. It can be used to improve the quality of products and services, and its application to the intelligent community, intelligent government affairs, intelligent education, intelligent management, people's livelihood, intelligent medical treatment, and other aspects of city production and life is realized, so that the development of all things is achieved and the harmony between the residents and the city is promoted, making the city more intelligent with the wisdom of the people there.

The digital information lays a good foundation for smart city construction and also entrusts with the new goal and new development vision in the new era. The management of a city is mainly based on computer technology, including IoT, blockchain technology, big data technology, and cloud computing technology. Through monitoring various aspects of the city, real-time data are obtained and provided for the decision-makers and administrative organizations of the city's management, avoiding blind decision-making and making city management more automatic and detailed [26].

Fig. 3. Distribution of pilot smart cities in the provinces of China at the end of November 2018

Smart city construction in China is supported by the central government and local state-owned organizations at all levels. From 2017 to 2018, the state issues a series of policies to encourage the development of smart transportation and healthcare based on cloud computing and artificial intelligence (AI). For example, in July 2017, the State Council issues "Development Planning of New Generation Artificial Intelligence", which points out six key tasks for developing AI in 2030, namely, vigorously developing the economy based on AI, improving the research ability of AI, building a sound research and production system and a new intelligent society based on AI, expanding the application of AI to social life, national defense and military, establishing a widely used application system of AI, and planning scientific research based on AI in the future [27]. Strengthening the application of AI to production, life, military and

other aspects of modern society can provide a good starting point for smart city construction. Figure 3 shows the distribution of pilot smart cities in the provinces of China at the end of November 2018 [28].

The above figure shows the current situations of smart cities in China by the end of 2018. The deeper the color is, the fewer pilots there are in the province. The survey shows that the cities along the southeast coast are the first pilots for reform. There are most pilots in Shandong, Jiangsu, and Hunan provinces in China, 30 in Shandong, 28 in Jiangsu, and 22 in Hunan [29].

Nowadays, smart city construction is extensively carried out, but there also appear some problems. First, the understanding of the connotation and development path of smart cities has not yet reached a consensus. Second, the connectivity and sharing of cross-industrial and cross-sectoral information become difficult, leading to the low interoperability between resources and businesses. The reasons may be that some units or departments take the public information as a means to charge the application cost produced by the implementation of the traditional charging policy, each department has its own information system, and the statistical data between the departments are inconsistent. Third, the organizations of each country cannot provide data to other working organs due to different reasons. It has been several years since the Central Committee introduces the concept of the triple play (telephone network, cable television, Internet) to the public. The data connectivity among the various working bodies is weak and overheating appears in smart city construction. Fourth, local governments in some areas are too busy to consider the local financial situation. Fifth, Internet construction is not perfect, and there exist hidden dangers in information security. In this case, many local governments follow the examples of other successful regions, resulting in the phenomenon of "thousands of cities are the same" [30]

Evaluation index system of smart cities. The evaluation index system of smart cities is shown in Table 2 [31, 32].

.**Table 2.** Evaluation index system of smart cities

Categories of indexes	Specific indexes
Economic development and future expectation F	Per capita GDP (Gross Domestic Product) F1 of the surrounding areas
	Regional real estate investment GDP share F2
	Per capita DI (Disposable Income) F3 of the residents in the city
	City consumer expenditure F4
Social development and future expectations S	The total regional population at the end of year S1
	Non-public road freight S2
	Non-public highway passenger volume S3
Public Service Quality and Future Development Expectations P	Per capita housing area P1
	The average number of beds per thousand patients P2 Average green area P3
	Per capita collection of public libraries P4
Technological innovation capability and the expectation I	The proportion of science and technology appropriations to local government expenditure I1
	The proportion of research and experimental research development funds to local production output I2
	Patent application I3
	Number of universities I4
	Number of college students I5
Network facility construction E	The proportion of infrastructure expenditure to GDP E1
	Thousands of people have network length E2
	Fixed number E3
	Number of mobile phones E4
	Number of network users E5

2.4. City brand building

Basic concept. City brand building is a comprehensive interdisciplinary subject that covers multiple disciplines. It takes brand science, urban economics, urban planning, advertising and marketing, and development as the core, and covers other disciplines, such as aesthetics, manufacturing, and history. Figure 4 shows the disciplines of city brand building [33, 34].

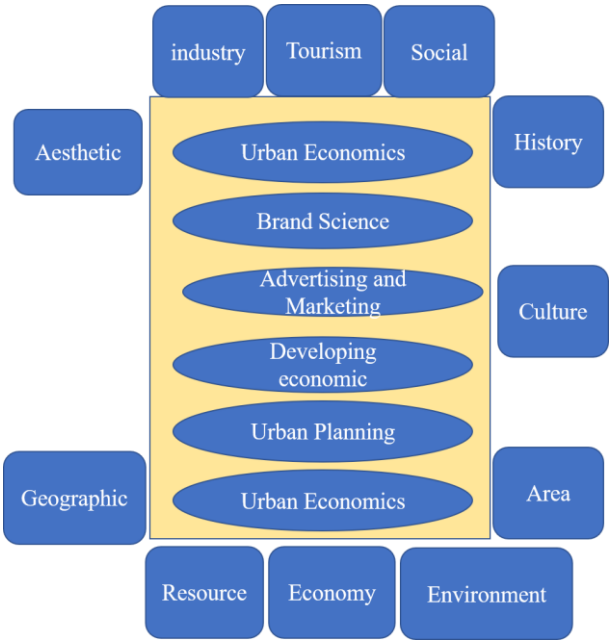


Fig. 4. Disciplines that city brand building covers

Basic theories of city brand building. Basic theories of city brand building include brand marketing theory, sustainable development theory, behavioral geography theory and urban planning theory. The core idea of brand marketing theory is to build a city brand with local characteristics, improve customers' loyalty and increase the added value of the city brand. Sustainable development theory is the principle of urban brand building, that is, it is people-oriented and a harmonious operating framework of "natural-economic-social" should be built. And behavioral geography can be a tool to help promote urban brand building. The urban planning theory is the basis of administrative behavior after urban construction. The city brand building theory is the extension of the urban planning theory and it is proposed to provide new ideas and perspectives for urban construction and development and enhance urban charm and expand the influence of city brands.

Strategies for city brand building. City brand building includes nine strategies, and their relations are shown in Figure 5.

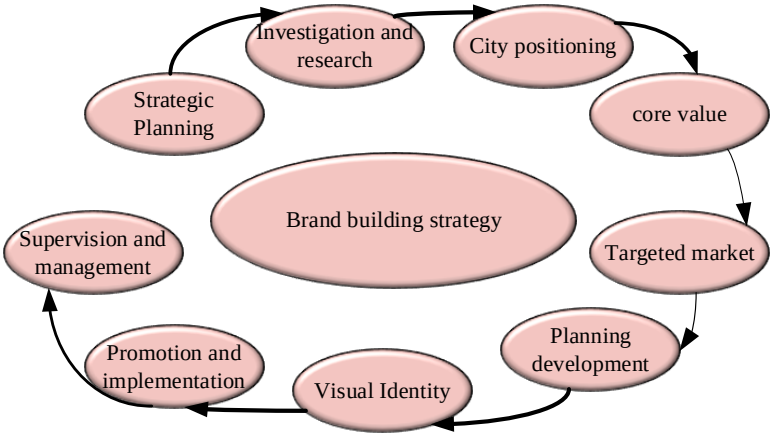


Fig. 5. Strategies for city brand building

Figure 5 shows that strategic planning is the first step in the whole process. The second step is to do some research. The third step is to excavate various resources of the city and find the personality, soul and concept of the city according to the principles of uniqueness, exclusiveness and authority. The fourth step is to build core values. The fifth step is to determine the target market. The sixth step is making plans for city development. The seventh step is determining the identifier. The eighth step is implementing the plan. The ninth step is supervising the promotion of a city brand.

Evaluation methods of city brands. The mainstream evaluation method of a city brand is through principal component analysis, which is a multivariate analysis method. The method can successfully detect the correlation of indexes and does not need to set the weight of the index. Therefore, the method is employed here.

3. Research Model and the Methodology

3.1. Experimental Procedure

In this study, the data of 15 cities including Beijing, Shanghai, Guangzhou, Shenzhen, Hangzhou, Nanjing, Changsha, Guizhou, Ningbo, Chongqing, Chengdu, Xi'an, Kunming, Zhengzhou and Jilin in 2020 are selected as the samples of this experiment. The evaluation model of smart city construction and development is tested experimentally. The Changsha city brand data in 2020 is selected as the sample for this study. Iteration is performed by software MATLAB (MATrix LABoratory), and a total of 20-dimensional vectors are output. The output layer is set to contain 20 nodes, the node is set to 1, and the hidden layer neuron range is controlled between 5-7, a total of 3 neural network structures.

Decision tree algorithm process: Let Q_X=(x=1,2,3,….,\|n|) be the proportion of x-type samples in sample D. When the value of Q_X is smaller, the purity of the D sample will be higher, and the Gini(D) will be smaller. The calculation of the information entropy Ent(D) and its Gini index Gini(D) is as follows:

$$Ent(D) = -\sum_{x=1}^{|n|} Q_X \log_2 Q_X \,. \tag{12}$$

$$Gini(D) = 1 - \sum_{x=1}^{|n|} {Q_X}^2 \,. \tag{13}$$

Data sharing is carried out through the code sharing platform GitLab to improve the repeatability of experiments. The specific steps are given as follows: (1) Enter the GitLab website to register an account. (2) Create a workgroup. (3) Create a GitLab project, and fill in the project name and its type, etc. (4) Import SSH Keys to generate Key file. (5) Upload the project.

3.2. GRA-BPNN model

GRA-BPNN model is realized by combining GRA method with BPNN to evaluate the development of smart cities. The implementation of the model based on the original GRA requires three steps, as shown in Table 3.

Table 3. The model added 3 steps based on the original GRA algorithm

Number	Steps
f	The sample is expressed as $x_{ml} = [x_{m1}, \ldots, x_{mn}]$, the expected output is $y_{ml} = [y_{m1}, \ldots, y_{mn}]$, m is the number of samples, l is the number of input vectors, and the initial value of the threshold is set.
g	If the output value of each layer is $O_{pi} = X_{pi}$, andX_{pi} represent the expected output value of each sample, the error between them is calculated.
h	Modify the weight ω_{ij} and threshold θ_{kj}, and the output layers t and k can be calculated by equations 12-15.
	$\omega_{ij}(t+1) = \omega_i(t) + \delta_{kj} o_{kj}$ (14)
	$\theta_{kj}(k+1) = \theta_{kj}(k) + \eta\delta_{ij}$ (15)
	$v_{kj}(t+1) = \theta_{pl}(t) + \delta_{pl}\rho_{pl}$ (16)
	$\theta_{pl}(k+1) = \theta_{pl}(k) + \eta\delta_{pl}$ (17)
	Based on the equations, the weighting coefficients δ, δ, o are improved, and the weighting coefficients of the output layer and the hidden layer are adjusted until the error is less than the preset error. The training is completed or the next cycle starts until the curve converges, and the results are obtained.

The basic structure of the algorithm is shown in Figure 6.

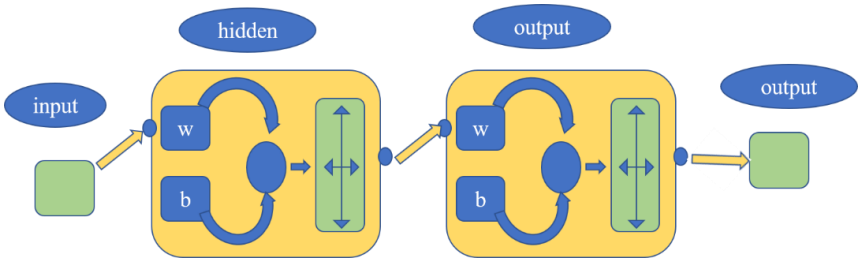


Fig. 6. Structure of the GRA-BPNN algorithm

3.3. GRA-BPNN model

Sample data. Beijing, Shanghai, Guangzhou, Shenzhen, Hangzhou, Nanjing, Changsha, Guizhou, Ningbo, Chongqing, Chengdu, Xi'an, Kunming, Zhengzhou, and Jilin are selected as the samples to test the evaluation model for smart city construction and development because they are the pilots in the national smart city construction. The 15 cities are the first cities that are declared as domestic smart cities, in which the infrastructures are relatively perfect, the level of economic development is high, and the supporting facilities for smart city construction are equipped. The data of the cities mainly come from "2020 national economic and social development statistics bulletin", "2020 scientific and technological progress statistical monitoring results and scientific and technological statistics bulletin", "2020 nineteen city financial technology investment comparison" and other statistics on the Internet. These smart cities are represented by English abbreviations to make the calculation process simple. They are represented by City1, City2, City3, City4, City5, City6, City7, City8, City9, City10, City11, City12, City13, City14 and City15. The original data of the construction and development of 15 smart cities in 2020 are shown in Table 4 below.

Experimental software. MATLAB is used to output 20-dimensional vectors. The output layer contains 20 nodes, and the output layer node is set to 1. The neurons in the hidden layer are adjusted from 5 to 7 according to the data, and a total of three structures of neural networks are obtained.

The entropy weight method (EWM), the grey correlation evaluation analysis method mentioned above and the evaluation method based on Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) are introduced to verify the performance of the GRA-BPNN model proposed. The evaluation results of the four different methods of smart city development are compared.

Table 4. Original data of the construction and development of 15 smart cities

	Indexes								
cities	F1	F2	F3	F4	S1	S2	S3	P1	P2
City 1	161409	0.472	4126.88	36947	1408.6	26859	65272	36.27	10.57
City 2	162499	0.372	44.962	29266	1538.0	85416	12802	34.5	9.41
City 3	134001	0.59	42840	25796	591.2	37256	27465	44.4	6.1
City 4	133890	0.280	41853	27839	1165.8	30728	212833	29.1	5.985
City 5	175907	0.36	39164.6	31601	833.4	89400	90380	35.5	8.6
City 6	87439	0.74	34067	23117	661.41	22709	37844	31.18	6.56
City 7	47229	1.050	19106	19106	463.30	22392	61540	23.78	7.3
City 8	62507	1.118	29194	21586	772.2	40242	41218	32.0	5.2
City 9	56372	0.852	21878	168.61	332.54	11642	4948	25.66	7.43
City 10	70476	0.233	31079	21473	1284.5	44439	135170	39.9	7.95
City 11	97197	0.42	39149	29420	740.68	28465	50257	38.2	3.83
City 12	121262	0.774	30932.3	21268	832.82	45640	30730	35.86	6.81
City 13	63952	0.96	29465	21794	554.59	29284	18498	31.73	8.91
City 14	182006	0.609	41110	26703	483.34	19509	27088.9	39.07	7.56
City 15	135693	0.768	40992	267.58	654.20	30110	16727	16727	6.51
	Indicators								
Cities	P3	P4	I1	I2	I3	I4	I5	E1	E2
City 1	16.6	1.12	5.54	0.172	51622	190	61.00	0.185	14.136
City 2	13.12	1.12	5.98	0.172	51622	69	51.588	0.085	2.900
City 3	11.99	3.14	4.02	0.148	49790	17	16	0.24	4.885
City 4	17.7	1.09	5.16	0.202	59517	6	8.35	0.229	2.116
City 5	16.6	0.97	3.99	0.145	59518	81	94.4	0.225	2.116
City 6	16.45	0.73	2.8	0.006	3278	111	17.77	0.287	3.216
City 7	12.3	1.34	1.06	0.328	4064	56	34.52	0.466	1.70
City 8	14.01	1.10	3.21	0.042	3642	61	45.45	0.328	0.942
City 9	11.08	2.99	2.017	0.002	20536	4	4.84	0.537	2.116
City 10	14.37	0.60	2.44	0.039	1123	81	97.75	0.177	4.06
City 11	11.9	0.51	2.19	0.026	3282	61	39.73	0.254	3.406
City 12	11.65	0.53	2.05	0.045	4432	54	71.28	0.591	4.576
City 13	11.9	0.94	50.4	0.038	34367	13	12.25	0.081	4.440
City 14	15.82	2.85	2.80	0.038	56105	55	81.85	0.252	4.441
City 15	16.1	2.40	1.9	0.029	1369	44	47.59	0.2577	1.803
Cities	E3	E4	E5						
City 1	67.98	174.5	42.42						
City 2	64.39	145.7	48.74						
City 3	54.20	140.2	49.96						
City 4	53.39	168.9	33.2						
City 5	71.28	150.1	79.57						
City 6	26.82	236.2	21.16						
City 7	23.64	214.9	21.8						
City 8	24.31	93.5	14.64						
City 9	41.68	194	33.19						
City 10	37.0	241.4	46.02						
City 11	33.18	74.0	24.48						
City 12	33.09	78.9	40.53						
City 13	43.89	131	51.36						
City 14	45.9	198.7	17.9						
City 15	27.95	157	50.42						

4. Results of Smart City Construction and Development and the Optimization of the City Brand Model Based on Neural Networks

4.1. Original Data of Smart City Construction and Development and their grey Correlation Degree

The original data are obtained based on the grey correlation theory, and the grey correlation degree of each index is shown in Table 5.

Table 5. Grey correlation degree of each index

Economic development and expectation F	Social development and expectations S	Public service quality and development expectations P	Technological innovation capability and expectation I	Network facilities construction E
F1=0.830	S1=0.779	P1=0.903	I1=0.784	E1=0.643
F2=0.769	S2=0.673	P2=0.891	I2=0.422	E2=0.519
F3=0.892	S3=0.613	P3=0.916	I3=0.706	E3=0.815
F4=0.856		P4=0.681	I4=0.542	E4=0.843
			I5=0.722	E5=0.706

The order of correlation in Table 5 is: P3 > P1 > F3 > P2 > F4 > E4 > F1 > E3 > I1 > S1 > F2 > I5 > E5 > I3 > P4 > S2 > E1 > I4 > E2 > S3 > I2. After the principle of correlation is simplified and the indexes that are not greater than 0.5 are excluded, 20 qualified indexes are finally obtained. Generally, the grey correlation index is between 0 and 1. The greater the correlation is, the greater the correlation between several indexes is. The above table shows that the correlation between development expectation P and smart city prosperity is the highest, and the other three indexes except P4 are about 0.9. The correlation between social development and expectation S is the lowest, about 0.7. The correlation between economic development and expected F is the second, indicating that public services and development expectations have the greatest impact on smart city development and city brand building. Financial funds also have a great impact on city development, while social development, expectations and network facilities have little impact on city development and brand building

The original data are normalized according to the obtained 20 indexes, and the results are shown in Figure 7.

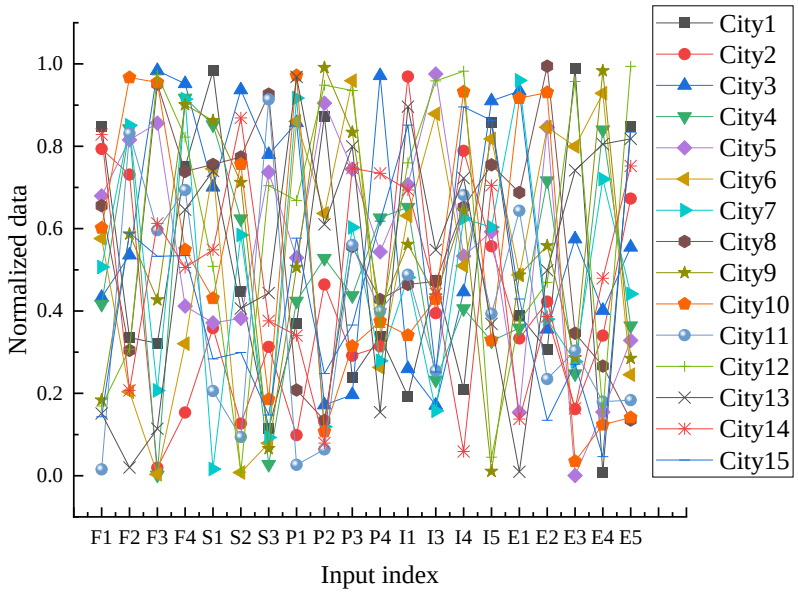


Fig. 7. Normalized results of original data

Figure 7 shows that the original data are normalized. According to the 20 available indexes, the samples are normalized by premnmx function, and the data after normalization are distributed between (-1, 1). The normalized data can make the function converge faster and improve the convergence accuracy.

4.2. Iterative results and the structure of networks

Figure 8 shows different iterative results with different numbers of the nodes in the hidden layer.

The blue line in the Figure represents the training iteration curve, the red line represents the test iteration curve, and the green line represents the verification iteration curve. When the number of hidden layer nodes of the neural network reaches 5, the best performance of the training reaches 1.63 * 10-2 after 20 iterations. And the curve converges and the number of hidden layer nodes of the neural network reaches 8 after13 training iterations. The optimal result of the training reaches 9.74 * 10–2 after 20 iterations. And the curve converges after the iterations are trained 2 times. The number of hidden layer nodes of the neural network reaches 7 after 0.131 * 102 iterations, and the best value during the training reaches 1.63 * 10-2. Since the number of iterations begins to decrease rapidly after the number of hidden layer nodes reaches more than 5, from 20 iterations of 5 nodes to 2, the curve converges, and the value reaches more than 9.74 * 10-2, indicating that the neural network model is good. Alhasoun and Gonzalez

(2019) proposed a method to collect and mark image data, and then train the deep Convolutional Neural Network (CNN) for street classification. A total of 9 nodes in 3 hidden layers is required to achieve the reduction from 30 iterations to 20 [35]. Cutchan et al. (2020) introduced a feature space for geospatial configuration, which contains semantic information. Then, the data in the feature space are used for deep learning. Finally, the high-precision prediction of city development is realized, and the number of iterations is reduced to 20 times by using eight neurons in two layers [36]. In short, the model based on BPNN and GRA has good performance.

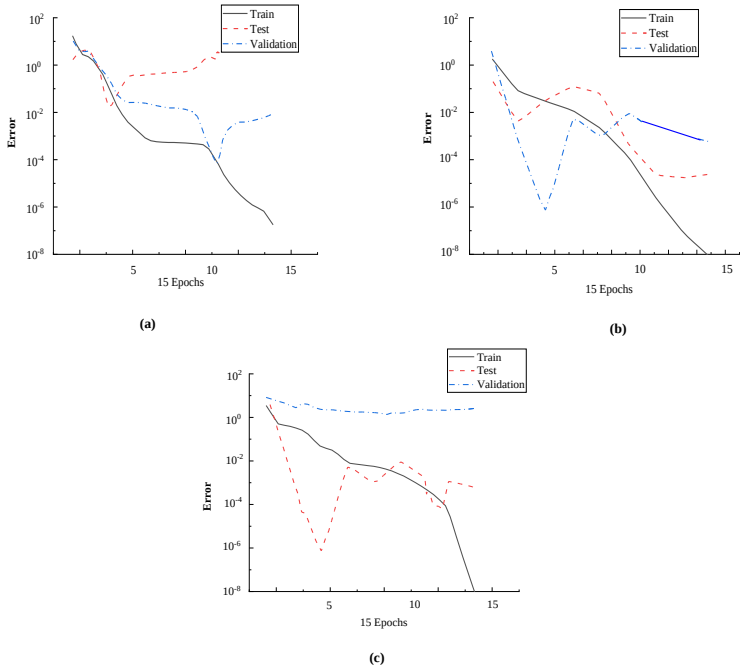


Fig. 8. Iterative results of different nodes in the hidden layer (a) *5 nodes* (b) *6 nodes* (c) *7 nodes*

The evaluation results of the 15 cities by the recommended model, EWM, GRA and the evaluation method based on TOPSIS are shown in Figure 9.

Figure 9 shows that the evaluation results of 15 cities by different algorithms are different. Among them, the EWM algorithm obtains the highest data value, the difference between the highest value and the lowest value is about 1, and the curve fluctuates significantly. The data values obtained by the GRA and TOPSIS algorithms are relatively close and the difference is not large, and the curves of the two algorithms fluctuate greatly. The data value obtained by the GRA-BPNN algorithm is the lowest, the overall evaluation fluctuates between 0.5 and 1, and the fluctuation range is the smallest, which is closer to the actual situation. The difference in the potential of the development level of smart cities is similar. However, the curve of the recommended model is relatively flat and stable than that of the other three methods, and its evaluation results are more reasonable than the other three methods, which proves that the evaluation method proposed is feasible and scientific. It further proves that the

evaluation model based on grey relational theory and BPNN is effective and reasonable in evaluating the development potential of smart city and plays a positive role in the future construction of smart city.

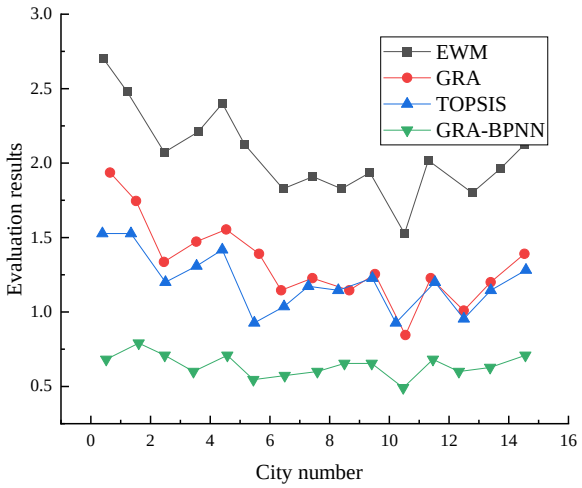


Fig. 9. Comparison of evaluation results of four algorithms

In summary, the proposed evaluation model based on grey correlation theory and BPNN has excellent performance, good stability and scientific test results. First, according to the grey correlation analysis method, the grey correlation degree of each index is calculated, and the original indexes are reduced. The reduced index is used as the input variable for BPNN to obtain an intelligent neural network for the development potential of smart cities based on GRA-BPNN. Finally, 15 smart cities in China are selected for empirical research. According to the designed intelligent neural network for the development potential of 15 cities based on GRA-BPNN, MATLAB 8.0 software is used as the calculation and simulation platform of the model to train and simulate the model. Finally, EWM, TOPSIS and grey correlation analysis method are used to evaluate the development potential of smart cities. The effectiveness and rationality of the evaluation model based on the grey correlation theory and BPNN are proved by comparing its evaluation results with other methods.

5. Conclusion

Based on smart city construction and development, the evaluation model of smart city construction and development based on BPNN is analyzed, and the strategies for smart city construction and city brand building are discussed. Then, the data of smart city construction and city brand building in various regions are analyzed and filtered, and the appropriate data are selected to establish the dataset. Subsequently, the evaluation

model for smart city construction and development based on BPNN is implemented and trained. Finally, the performance of the evaluation model is tested by comparing the evaluation results of EWM, GRA and TOPSIS. The experimental results show that the performance of the evaluation model for smart city construction and city brand building based on BPNN is good, and the accuracy and timeliness of evaluation are achieved. Although some achievements are made, there are still some limitations that have to be improved. And they are summarized as follows: (1) in the design of the evaluation index system of smart city development, there are few research results to be referred to, and the designed evaluation index system of smart city development needs to be further improved; (2) in terms of empirical research, only 15 pilot cities are selected as research samples. The size of the samples is small, and the conclusion may not be universal; (3) in-depth analysis of smart city construction is not comprehensive. In future research, the above three points should be improved. In addition, the following two will be the focuses of the follow-up research: (1) improving the compatibility of the model and optimizing the algorithm, and adding more machine learning algorithms to improve the repeatability of model results and increase model reliability. (2) Coordinating all parties to carefully select samples. (3) Getting more training set samples to train the model.

Acknowledgment. This work was supported by the High-Level Talents Research Fund Project (2019gcky05). This work was also financially supported by China Scholarship Council (CSC) fund.

References

1. Sol, D., Maspons, J., Gonzalez-Voyer, A., Morales-Castilla, I., Garamszegi, L. Z., Møller, A. P.: Risk-taking behavior, urbanization and the pace of life in birds. Behavioral Ecology and Sociobiology, Vol. 72, No. 3, 1-9. (2018)
2. Satterthwaite, D., McGranahan, G., & Tacoli, C.: Urbanization and its implications for food and farming. Philosophical transactions of the royal society B: biological sciences, Vol. 365, No. 1554, 2809-2820. (2010)
3. Lytras, M. D., Visvizi, A.: Who uses smart city services and what to make of it: Toward interdisciplinary smart cities research. Sustainability, Vol. 10, No. 6, 1998. (2018)
4. Allam, Z., Newman, P.: Redefining the smart city: Culture, metabolism and governance. Smart Cities, Vol. 1, No. 1, 4-25. (2018)
5. Xiang, Y., Cui, Y., Wei, Y.: A study on the construction of evaluative system of Smart City based on entropy-weighted matter-element model method. Concurrency and computation: practice and experience, Vol. 31, No. 9, e5010. (2019)
6. Su, L., Fan, J., Fu, L.: Exploration of smart city construction under new urbanization: A case study of Jinzhou-Huludao Coastal Area. Sustainable Computing: Informatics and Systems, Vol. 27, 100403. (2019)
7. Ming, Y., Wang, E.: Identity-based encryption with filtered equality test for smart city applications. Sensors, Vol. 19, No. 14, 3046. (2019)
8. Vekinis, A. A., Constantoudis, V.: Quantifying geometric tip-sample effects in AFM measurements using certainty graphs. Micro and Nano Engineering, Vol. 8, No. 3, 57-61. (2020)
9. Dubourg, F., G., Rebergen, D., Parsons, C.: 1438PSomaticNET: Neural network evaluation of somatic mutations in cancer. Annals of Oncology. Vol. 30, No. 5, 584-586. (2019)

10. Ostad-Ali-Askari, K., Shayannejad, M.: Computation of subsurface drain spacing in the unsteady conditions using Artificial Neural Networks (ANN). Applied Water Science, Vol. 11, No. 2, 21-30. (2021)
11. Ostad-Ali-Askari, K., Shayannejad, M.: Ghorbanizadeh-Kharazi H. Artificial Neural Network for Modeling Nitrate Pollution of Groundwater in Marginal Area of Zayandeh-Rood River, Isfahan, Iran. KSCE Journal of Civil Engineering, Korean Society of Civil Engineers, Vol. 21, No. 1, 134-140. (2017)
12. Massimo, A., Morten, N.: Gapped sequence alignment using artificial neural networks: application to the MHC class I system. Bioinformatics, Vol. 6, No. 4, 511-552. (2018)
13. Chen, Y.: Application of the best evacuation model of deep learning in the design of public structures. Image and Vision Computing, Vol. 102, No. 1, 975-977. (2020)
14. Chang, H.: Applying an Artificial Neural Network to Pattern Recognition in Fabric Defects. Textile Research Journal, Vol. 7, No. 2, 372-399. (2018)
15. Yan, F., Lin, Z., Wang, X.: Evaluation and prediction of bond strength of GFRP-bar reinforced concrete using artificial neural network optimized with genetic algorithm. Composite Structures, Vol. 161, No. 1, 441–452. (2017)
16. Hubeatir, K.: Laser transmission welding of PMMA using IR semiconductor laser complemented by the Taguchi method and grey relational analysis - ScienceDirect. Textile Research Journal, Vol. 4, No. 2, 472-480. (2020)
17. Topal, M., Yaanolu, A.: Comparison of quality characteristics in honey using grey relational analysis and principal component analysis methods. Journal of Animal and Plant Sciences, Vol. 28, No. 1, 252-263. (2018)
18. Anand, G., Alagumurthi, N., Elansezhian, R.: Investigation of drilling parameters on hybrid polymer composites using grey relational analysis, regression, fuzzy logic, and ANN models. Journal of the Brazilian Society of Mechanical Sciences and Engineering, Vol. 40, No. 4, 214-215. (2018)
19. Das, P., Chakraborty, S.: Application of grey correlation-based EDAS method for parametric optimization of non-traditional machining processes. Scientia Iranica, Vol. 2, No. 1, 25-28. (2020)
20. Cholilie, I., Rahmat, A., Gabriel, A.: Sorghum Crackers (Sorghum bicolor L.) as an Effort in Utilizing Food Material of Local Wisdom in Lamongan City. Gontor agrotech Science Journal, Vol. 6, No. 3, 283. (2020)
21. Kassensnoor, E .: 'Failure to adjust: Boston's bid for the 2024 Olympics and the difficulties of learning Olympic wisdom. Environment and planning, Vol. 51, No 8, 1684-1702. (2019)
22. Mengelkamp, E., Notheisen, B., Beer, C.: A blockchain-based smart grid: towards sustainable local energy markets. Computer Science - Research and Development, Vol. 33, No. 1, 207-214. (2018)
23. Priyanka, M.: An Expert model for DNA Based Encryption Technology using Cloud Computing. International Journal of Advanced Trends in Computer Science and Engineering, Vol. 8, No. 1, 15-18. (2019)
24. Singh, S., Yassine, A.: Big Data Mining of Energy Time Series for Behavioral Analytics and Energy Consumption Forecasting. Energies, Vol. 11, No. 2, 452-453. (2018)
25. Ronen, E., Shamir, A.: Weingarten A, et al. IoT Goes Nuclear: Creating a Zigbee Chain Reaction. IEEE Security & Privacy, Vol. 16, No. 1, 54-62. (2018)
26. Steiner, F.: The ecological wisdom of plan-making. Journal of Urban Management, Vol. 7, No. 3, 124-130. (2018)
27. Michelle, V., David, K.: Digitising the wisdom of our elders: connectedness through digital storytelling. Agng and Society, Vol. 2, No. 1, 1-21. (2018)
28. Hok, B., Lena, D.: Not only Eating Together: Space and Green Social Work Intervention in a Hazard-Affected Area in Ya'an, Sichuan of China. British Journal of Social Work, Vol. 48, No. 5, 1409-1431. (2018)
29. Zhang, M.: International Conference on Intelligent Transportation. Big Data & Smart City (ICITBS) - Reform of Teaching Method, Vol. 2, No. 1, 187-190. (2018)

30. Ronarizkia, A., Giriwati, N.: Visual Character of Colonial Building Facade in Suroyo Street Corridor, Probolinggo City Indonesia. Local Wisdom Jurnal Ilmiah Kajian Kearifan Lokal, Vol. 12, No. 1, 12-19. (2020)
31. Gall, R., Jabot, F.: Health impact assessment on urban development projects in France: finding pathways to fit practice to context. Global Health Promotion, Vol. 24, No. 2, 25-27. (2018)
32. Gammaz, S., Hagras, M.: Social impact assessment as a tool for achevie social sustainability in urban development. Journal of Engineering and Applied Science, Vol. 67, No. 7, 1669-1687. (2020)
33. Woo, K., Khoo, S.: Ecology and new urban program: A case study of Penang state own brand of affordable housing program - ScienceDirect. Journal of Urban Management, Vol. 9, No. 2, 168-179. (2020)
34. Cobbinah, P., Kosoe, E., Diawuo, F.: Environmental planning crisis in urban Ghana: Local responses to nature's call. The Science of the Total Environment, Vol. 701, No. 20, 1-11. (2020)
35. Alhasoun, F., Gonzalez, M.: Streetify: Using Street View Imagery and Deep Learning for Urban Streets Development, Vol. 9, No. 18, 461-464. (2019)
36. Cutchan, M. M., Zdal, K. S., Giannopoulos I. Semantic-based urban growth prediction Transactions in GIS, Vol. 6, No. 10, 251-255. (2020)

Yingji Li was born in Qiqihaer , Heilongjiang , P.R. China, in 1985. He received the doctoral degree from Daejeon University, South Korea. Now, he works in School of Humanities and Management, Yunnan University of Chinese Medicine. His research interest include behavior analysis and information system management.

Yufeng Qian was born in Wuhan, Hubei, China, in 1986. He received the B.S. degree in information and computing science from Central China Normal University, Wuhan, China, the M.S. degree and the Ph.D. degree in computational mathematics from Wuhan University, Wuhan, China, in 2008, 2010 and 2013, respectively. Currently, he works as a lecturer in the School of Science, Hubei University of Technology, Wuhan, China. His research interests include applied mathematics, complex systems and complex networks.

Qiang Li was born in Weifang, Shandong. P.R. China, in 1990.He received the master's degree from Daejeon University, South Korea.Now, he works in School of Economics and Management, Shanghai Technical Institute of Electronics & Information. His research interest include teaching management, business management and data analysis.

Linna Li was born in Wuhan, Hubei, P.R. China, in 1992. She received the bachelor's degree from Wuhan University, P.R. China and the Master degree from Chongqing University, P.R. China. She is currently working toward the Ph.D. degree in the University of Melbourne. Her research interests include smart city, sustainable development in urban and rural settlements.

Received: July 15, 2022; Accepted: November 12, 2022.

Computer Science and Information Systems 20(2):595–0000 https://doi.org/10.2298/CSIS220712011L

Security Performance Analysis of Active Intelligent Reflective Surface Assisted Wireless Communication

Yiming Li[1], Xitao Liang[1], Wenwu Xie[1,⋆], and Juan Zhu[2]

[1] Hunan Institute of Science and Technology, Department of Information and Communication Engineering Hunan Province, China
gavinxie@hnist.edu.cn
[2] School of Physics & Electronic Engineering, Hubei University of Arts and Science, Xiang Yang, Hubei Province, China
journey1022@126.com

Abstract. As a new communication technology, Intelligent Reflecting Surface (IRS) can intelligently reconfigure the wireless propagation environment by integrating many passive/active reflective elements on the plane. According to the characteristics that IRS can adjust the propagation channel intelligently, this paper applies IRS to wireless security communication, and studies how to make the security rate reach the optimal security capacity from the perspective of optimization technology. In this paper, two schemes of passive/active IRS are considered, and the corresponding safety rate maximization algorithm is proposed. In view of the non-convexity of the objective function, on the one hand, in the passive IRS scheme, the Dinkelbach method is used to transform the objective function into a form that is easy to handle, and the original problem is transformed into a convex problem through the continuous convex approximation method; On the other hand, under the active IRS scheme, aiming at the complexity of the original problem, the first order Taylor expansion is used to obtain the lower bound of the optimization problem, and a minimax optimization algorithm is proposed. Finally, the performance of the proposed algorithm is verified by simulation. The simulation results show that the algorithm designed with active IRS has better security rate than the algorithm designed with passive IRS under the same parameter settings.

Keywords: IRS, Physical layer Secrecy, Convex optimization.

1. Introduction

Intelligent Reflecting Surfaces (IRSs) are man-made surfaces of electromagnetic(EM) material controlled by integrated electronic devices, which have attracted widespread attention due to their unique wireless communication capabilities[1]. The development of IRS has evolved from related researches on metamaterials and metasurfaces, and current implementations include traditional reflectarray, liquid crystal surfaces, and software-defined primitive surfaces [2,3,4]. With the emergence of IRS objects, IRS technology provides a new idea for improving the channel capacity of wireless communication. Under the premise that IRS can be realized, more and more researchers analyze the positive effects of IRS on wireless communication from different research fields, such as performance optimization, channel estimation, security, etc [5].

⋆ Corresponding author

Specifically, the passive IRS can adjust the phase of the incident electromagnetic wave by controlling the components deployed on the IRS, thereby controlling the direction of the electromagnetic wave at the receiver. Compared with the traditional decoding and forwarding relaying technology, passive IRS can shape the wireless signals through software programming without additional energy sources, increasing the channel capacity without consuming spectrum resources. Most of the current research shows that passive IRS can improve the channel capacity of target users [6,7,8,9], but these studies ignore the "double fading" effect brought by passive IRS, and only consider that the direct link is very poor atypical scenario. Therefore, in order to overcome the basic limitation of the "double fading" effect, some researchers proposed the application of active IRS in wireless communication systems [12,13,14,15,16,17,18]. Unlike passive IRS, which passively reflects signals without amplification, active IRS can not only adjust the phase of incident electromagnetic waves, but also amplify reflected signals. However, the noise in the signal transmission is also amplified accordingly, and additional power is required. However, In comparison with the amplify-and-forward relay technology, active IRS does not require encoding and decoding operations. Reference 11 specifically expounds the concept of active IRS, its hardware structure and signal model, introduces how active IRS amplifies the incident signal, and compares and analyzes the channel capacity gain brought by active IRS and passive IRS through experimental measurements. Reference 12 introduces an active IRS into a single-input multiple-output system, and maximizes the received signal power of the system and minimizes the IRS-related noise at the receiver by optimizing the reflection coefficient matrix of the IRS and the receiving beamforming at the receiver. The results prove that the active IRS- assisted system has better performance than the traditional passive IRS- assisted system under the same power budget. However, if the two systems have the same total power budget and the number of reflection units of the IRS is sufficient, the channel capacity brought by the passive IRS is better than that of the active IRS[13]. References 14,15,16 also demonstrate the superiority of active IRS over passive IRS. It is worth noting that, unlike the existing passive IRS, which passively reflects the signal without amplitude amplification, the active IRS requires extra power to support the active amplification of the reflected signal, so with the same number of reflective elements, Active IRS requires a larger power budget than passive IRS [17]. In order to reduce the power loss of active IRS, the literature 18 proposes a sub-connection architecture of active IRS. At the cost of reduced degrees of freedom required for beamforming design, the components on the IRS independently control their phase shifts, but share the same power amplifier, the simulation results demonstrate that the sub-connection structure is an energy-efficient implementation of the active IRS. More importantly, the team of Mr. Dai from Tsinghua University has a mature IRS hardware design method[19], which has promoted more and more researchers to study the beneficial effects of IRS on wireless communication from different fields.

With the rapid development of the Internet of Things and the continuous upgrading of various eavesdropping methods, the disadvantages brought by traditional security technologies are also increasing. The emergence of IRS technology provides a new direction for PLS research. Although PLS for different wireless systems has been extensively studied in many literatures [20,21,22,23], the application of IRS to improve PLS is still not well studied. Through literature review, it is found that most authors consider passive IRS to improve the PLS of the system. As in the literature

24,25,26,27,28,29,32,33,34,35,36,37,38,39, the security performance of a passive IRS-assisted system is considered, and the security capacity is maximized by optimizing the reflection phase shift and/or transmission probability of the passive IRS. Few literatures consider the application of active IRS. Furthermore, existing research on active IRS aspects is limited to non-secure aspects, that is, eavesdropping users are not considered in the system model. Although active IRS greatly improves the communication quality of users, it will also leak more information to eavesdropping users due to the broadcast nature of wireless channels. Therefore, it is also necessary to consider the safety performance gain brought by active IRS.

2. System Model

In order to reduce the influence of "double fading", this section further considers the application of active IRS in PLS, and the considered system model is shown in Fig. 1. In the active IRS-assisted communication system model, h_D, h_E are the channels from the base station to the legitimate user and the base station to the eavesdropping user, respectively. $\mathbf{h}_{br} \in \mathbb{C}^{N\times 1}$, $\mathbf{h}_{rd} \in \mathbb{C}^{N\times 1}$ are the channels between the base station to the IRS and the IRS to the legitimate user, respectively, $\mathbf{h}_{re} \in \mathbb{C}^{N\times 1}$ is a channel from the IRS to the eavesdropping user. In addition, considering that the base station knows the CSI of all users, it is assumed that the eavesdropping user passively eavesdrops on the information transmitted by the base station to the legitimate users, without causing any interference to the legitimate users.

The reflection coefficient diagonal matrix of IRS is defined as $\Theta = diag(a_1 e^{j\theta_1}, \ldots, a_N e^{j\theta_N})$, a_n and $\theta_n \in [0, 2\pi]$ are the amplification factor matrix and phase shift matrix of IRS, respectively. In particular, the active IRS can not only change the phase shift of the channel, but also amplify the signal and then reflect it to the user[12], So the magnification factor $a_n \geq 1, \forall n$. However, the active IRS requires additional power to support the amplification of the reflected signal. In addition, active IRS amplifies the useful signal while the amplified noise signal can not be ignored.. Therefore, the received signals at the legitimate user and the eavesdropping user can be respectively expressed as[31]

$$y_s = \sqrt{p_t}\left(h_D + \mathbf{h}_{rd}^H \Theta \mathbf{h}_{br}\right) x_s + \underbrace{\mathbf{h}_{rd}^H \Theta n_r}_{\substack{\text{noise by} \\ \text{active RIS}}} + n_s \tag{1}$$

$$y_s = \sqrt{p_t}\left(h_D + \mathbf{h}_{re}^H \Theta \mathbf{h}_{br}\right) x_s + \underbrace{\mathbf{h}_{re}^H \Theta n_r}_{\substack{\text{noise by} \\ \text{active RIS}}} + n_e \tag{2}$$

where n_r is the noise introduced for active IRS amplification, and $n_r \sim \mathcal{CN}(0, \sigma_r^2)$. Then, the channel capacities of the legitimate end and the eavesdropping end are respectively defined as

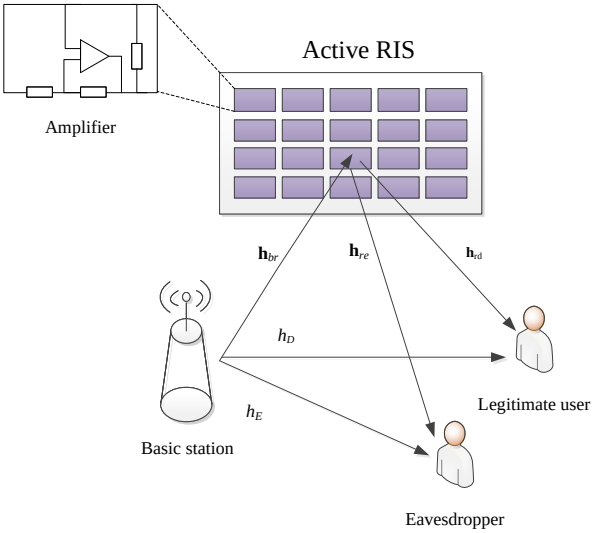


Fig. 1. Active IRS-assisted Communication system model

$$C_S = \log\left(1 + \frac{p_t \left|h_D + \mathbf{h}_{rd}^H \Theta \mathbf{h}_{br}\right|^2}{\left\|\mathbf{h}_{rd}^H \Theta\right\|^2 \sigma_r^2 + \sigma_s^2}\right) \tag{3}$$

$$C_E = \log\left(1 + \frac{p_t \left|h_E + \mathbf{h}_{re}^H \Theta \mathbf{h}_{br}\right|^2}{\left\|\mathbf{h}_{re}^H \Theta\right\|^2 \sigma_r^2 + \sigma_e^2}\right) \tag{4}$$

As mentioned above, it should be noted that, compared to passive IRS, active IRS needs to consider the amplified noise. To ensure that legitimate users can correctly decode the signal sent by the base station, the noise introduced by the active IRS should satisfy the following constraints

$$\left\|\Theta(\phi)\mathbf{h}_{br}\right\|^2 + \left\|\Theta(\phi)\mathbf{I}_N\right\|^2 \bar{\gamma}_r \leq \bar{P}_A \tag{5}$$

where $\bar{P}_A = P_A/p_t$, $\bar{\gamma}_r = \sigma_r/p_t$, P_A is the maximum noise amplification power budget for active IRS, $\mathbf{I}_N$ is the identity matrix of dimension N. In addition, the amplification gain of the active IRS also needs to satisfy the upper bound constraint. Let $a_{\max}$ be the maximum gain of amplification, assuming that the maximum gain of each reflector of each IRS is the same. Then, the reflection coefficient constraint of the IRS can be expressed as

$$|\phi_i| \leq a_{i,\max}, i = 1, \cdots, N \tag{6}$$

where $\phi = \mathrm{diag}(\Theta) = \left[a_1 e^{j\theta_1}, \cdots, a_N e^{j\theta_N}\right]^H$. According to the above constraints, it is necessary to optimize the phase shift and amplification gain to maximize the safe rate of the system. The optimization problem can be expressed as

$$\begin{aligned} \mathrm{P2}: \max_{\phi} \log\left(1+\frac{\left|h_D+\phi^H\mathbf{h}_d\right|^2}{\left|\phi^H \mathrm{diag}\left(\mathbf{h}_{rd}^H\right)\right|^2\bar{\gamma}_r+\bar{\gamma}_s}\right)- \\ \log\left(1+\frac{\left|h_E+\phi^H\mathbf{h}_e\right|^2}{\left|\phi^H \mathrm{diag}\left(\mathbf{h}_{re}^H\right)\right|^2\bar{\gamma}_r+\bar{\gamma}_e}\right) \end{aligned} \tag{7}$$

$$\text{s.t. } |\phi_i| \le a_{\max}, i = 1, \cdots, N \tag{8}$$

$$\left\|\phi^H \mathrm{diag}\left(\mathbf{h}_{br}\right)\right\|^2 + |\phi|^2\bar{\gamma}_s \le \bar{P}_A \tag{9}$$

where $\bar{\gamma}_i = \sigma_i^2/p_t, i \in \{r, s, e\}$, $\mathbf{h}_i = \mathrm{diag}\left(\mathbf{h}_{ri}^H\right)\mathbf{h}_{br}, i \in \{d, e\}$. The objective function and constraints in problem P2 are both non-convex, so they cannot be solved directly. In order to solve this problem, a joint optimization algorithm based on SDR method and min-max algorithm is proposed. The algorithm to solve this problem is introduced below.

3. Optimal Solution

Since the numerator and denominator of the objective function in problem P2 and all the constraints are related to the optimization variables ϕ, the optimization solution cannot be directly solved. First, let $\mathrm{V} = [\phi; 1]\left[\phi^H, 1\right]$, rewrite the objective function in P2. The channel capacity of the legal end can be reformulated as

$$\begin{aligned} &\log\left(1+\frac{\left|h_D+\phi^H\mathbf{h}_d\right|^2}{\left|\phi^H \mathrm{diag}\left(\mathbf{h}_{rd}^H\right)\right|^2\bar{\gamma}_r+\bar{\gamma}_d}\right) \\ &= \log\left(\left|\phi^H \mathrm{diag}\left(\mathbf{h}_{rd}\right)\right|^2\bar{\gamma}_r+\bar{\gamma}_d+\left|h_D+\phi^H\mathbf{h}_d\right|^2\right) \\ &- \log\left(\left|\phi^H \mathrm{diag}\left(\mathbf{h}_{rd}^H\right)\right|^2\bar{\gamma}_r+\bar{\gamma}_d\right) \\ &= \log\left(\mathrm{tr}\left(\mathrm{H}_D\ \mathrm{V}\right)\right) - \log\left(\mathrm{tr}\left(\tilde{\mathrm{H}}_D\ \mathrm{V}\right)\right) \end{aligned} \tag{10}$$

where $\mathrm{H}_D = \begin{bmatrix} \left(\mathrm{diag}\left(\mathbf{h}_{rd}\right)\mathrm{diag}\left(\mathbf{h}_{rd}\right)^H\bar{\gamma}_r+\mathbf{h}_d\mathbf{h}_d^H\right), \mathbf{h}_d h_D^H \\ h_D\mathbf{h}_d^H, \bar{\gamma}_d + h_D h_D^H \end{bmatrix}$, $\tilde{\mathrm{H}}_D = \begin{bmatrix} \mathrm{diag}\left(\mathbf{h}_{rd}\right)\mathrm{diag}\left(\mathbf{h}_{rd}\right)^H\bar{\gamma}_r, 0_{N\times 1} \\ 0_{1\times N}, \bar{\gamma}_d \end{bmatrix}$. Similarly, the channel capacity of the eavesdropping side can be expressed as

$$\log\left(1+\frac{\left|h_E+\phi^H\mathbf{h}_e\right|^2}{\left|\phi^H \mathrm{diag}\left(\mathbf{h}_{re}^H\right)\right|^2\bar{\gamma}_r+\bar{\gamma}_e}\right) = \log\left(\mathrm{tr}\left(\mathrm{H}_E\ \mathrm{V}\right)\right) - \log\left(\mathrm{tr}\left(\tilde{\mathrm{H}}_E\ \mathrm{V}\right)\right) \tag{11}$$

where, $\mathrm{H}_E = \begin{bmatrix} \left(\mathrm{diag}\left(\mathbf{h}_{re}\right)\mathrm{diag}\left(\mathbf{h}_{re}\right)^H\bar{\gamma}_r+\mathbf{h}_e\mathbf{h}_e^H\right), \mathbf{h}_e h_E^H \\ h_E\mathbf{h}_e^H, \bar{\gamma}_e + h_E h_E^H \end{bmatrix}$, $\tilde{\mathrm{H}}_E = \begin{bmatrix} \mathrm{diag}\left(\mathbf{h}_{re}\right)\mathrm{diag}\left(\mathbf{h}_{re}\right)^H\bar{\gamma}_r, 0_{N\times 1} \\ 0_{1\times N}, \bar{\gamma}_e \end{bmatrix}$. Then, let $\Pi =$

$\begin{bmatrix} p_t \operatorname{diag}(\mathbf{h}_{br}) \operatorname{diag}(\mathbf{h}_{br})^H + \sigma_r^2 \mathbf{I}_N, 0_{N\times 1} \\ 0_{1\times N}, 0 \end{bmatrix}$, Constraints (9) can be expressed as

$$\operatorname{tr}(\bar{\Pi}\mathrm{V}) \le \bar{P}_A \tag{12}$$

Therefore, the optimization problem P2 can be reformulated as

$$\begin{aligned} \mathrm{P2}a: \max_{\mathrm{V}} C(\mathrm{V}) = \log(\operatorname{tr}(\mathrm{H}_D\ \mathrm{V})) - \log\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_D\ \mathrm{V}\right)\right) - \\ \log(\operatorname{tr}(\mathrm{H}_E\ \mathrm{V})) + \log\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_E\ \mathrm{V}\right)\right) \end{aligned} \tag{13}$$

$$\text{s.t. } \mathrm{V}(i,i) \le a_{\max}, i = 1, \cdots, N \tag{14}$$

$$\mathrm{V}(N+1, N+1) = 1,\ \mathrm{V} \ge 0 \tag{15}$$

$$\operatorname{tr}(\bar{\Pi}\mathrm{V}) \le \bar{P}_A \tag{16}$$

After processing, it is found that the objective function of problem P2a is still non-convex. We consider approximating the objective function, and then optimize the approximated objective function through a minimum-maximization algorithm. According to the first-order Taylor expansion, any concave function $f(\mathrm{x})$, for points within the domain $\mathrm{x}, \tilde{\mathrm{x}}$, can be written as $f(\mathrm{x}) \le f(\tilde{\mathrm{x}}) + (\nabla f(\tilde{\mathrm{x}}))^T(\mathrm{x}-\tilde{\mathrm{x}})$. Any concave function $g(X)$, for points within the domain $\mathrm{X}, \tilde{\mathrm{X}}$, can be written as $g(\mathrm{X}) \le g(\tilde{\mathrm{X}}) + \operatorname{tr}(\nabla g(\tilde{\mathrm{X}})(\mathrm{X}-\tilde{\mathrm{X}}))$.Then Since $\operatorname{tr}\left(\tilde{\mathrm{H}}_D\ \mathrm{V}\right)$ is a linear function of V, $\log_2\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_D\ \mathrm{V}\right)\right)$ is a convex function of V. Therefore, for any feasible point $\tilde{\mathrm{V}}$, it is satisfied $\log_2\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_D\ \mathrm{V}\right)\right) \le \log_2\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_D\tilde{\mathrm{V}}\right)\right) + \operatorname{tr}\left(\tilde{\mathrm{H}}_D / \operatorname{tr}\left(\tilde{\mathrm{H}}_D\tilde{\mathrm{V}}\right)(\mathrm{V}-\tilde{\mathrm{V}})\right)$. Similarly, for any feasible point , it is also satisfied $\log_2(\operatorname{tr}(\mathrm{H}_E\ \mathrm{V})) \le \log_2\left(\operatorname{tr}\left(\mathrm{H}_E\tilde{\mathrm{V}}\right)\right) + \operatorname{tr}\left(\mathrm{H}_E / \operatorname{tr}\left(\mathrm{H}_E\tilde{\mathrm{V}}\right)(\mathrm{V}-\tilde{\mathrm{V}})\right)$. Therefore, the lower bound of the objective function of problem P2a can be expressed as

$$\begin{aligned} C(\mathrm{V}) \ge \log(\operatorname{tr}(\mathrm{H}_D\ \mathrm{V})) + \log\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_E\ \mathrm{V}\right)\right) - \\ \log_2\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_D\tilde{\mathrm{V}}\right)\right) - \operatorname{tr}\left(\tilde{\mathrm{H}}_D / \operatorname{tr}\left(\tilde{\mathrm{H}}_D\tilde{\mathrm{V}}\right)(\mathrm{V}-\tilde{\mathrm{V}})\right) - \\ \log_2\left(\operatorname{tr}\left(\mathrm{H}_E\tilde{\mathrm{V}}\right)\right) - \operatorname{tr}\left(\mathrm{H}_E / \operatorname{tr}\left(\mathrm{H}_E\tilde{\mathrm{V}}\right)(\mathrm{V}-\tilde{\mathrm{V}})\right) \end{aligned} \tag{17}$$

Then, by replacing the objective function in P2a with equation (17), and removing the constant term, the optimization problem can be approximately expressed as

$$\begin{aligned} \mathrm{P2}b: \max_{\mathrm{V}} \log(\operatorname{tr}(\mathrm{H}_D\ \mathrm{V})) + \log\left(\operatorname{tr}\left(\tilde{\mathrm{H}}_E\ \mathrm{V}\right)\right) - \\ \operatorname{tr}\left(\tilde{\mathrm{H}}_D / \operatorname{tr}\left(\tilde{\mathrm{H}}_D\tilde{\mathrm{V}}\right)\mathrm{V}\right) - \operatorname{tr}\left(\mathrm{H}_E / \operatorname{tr}\left(\mathrm{H}_E\tilde{\mathrm{V}}\right)\mathrm{V}\right) \end{aligned} \tag{18}$$

$$\text{s.t. } \mathrm{V}(i,i) \le a_{\max}, i = 1, \cdots, N \tag{19}$$

$$\mathrm{V}(N+1, N+1) = 1,\ \mathrm{V} \ge 0 \tag{20}$$

$$\operatorname{tr}(\bar{\Pi}\mathrm{V}) \le \bar{P}_A \tag{21}$$

The problem P2b is already a convex problem and can be solved with the CVX toolbox. After the matrix V is obtained after optimization, if the rank of the matrix V is not

1, it can be restored to the rank 1 matrix by Gaussian randomization. Then, according to $\Theta = \mathrm{diag}\left(u_{\max}(\mathrm{V})\sqrt{\lambda_{\max}(\mathrm{V})}\right)[1:N]$, the original phase shift matrix Θ can be obtained, where $\mu_{\max}(\bullet)$ is the eigenvector corresponding to the largest eigenvalue of the matrix, and $\lambda_{\max}(\bullet)$ is the largest eigenvalue of the matrix. The entire optimization algorithm flow is shown in the algorithm Table 1.

Table 1. Algorithms for problem P2

Algorithm 1 Algorithm for Problem P2
1: Initialize feasible point $\tilde{\mathrm{V}}$, convergent value ε
2:Calculate the value of $C(\tilde{\mathrm{V}})$,set $C(\mathrm{V}) = C(\tilde{\mathrm{V}}) + 2\varepsilon\mathbf{I}$
3:**While** $\left\|C(V) - C(\tilde{V})\right\| \geq \varepsilon$:
4: Solving the optimization problem P2b gets ;
5: **if not** $rank(V) = 1$ **then**
6: use the Gaussian randomization method;
7: **end if**
8: Calculate the value of $C(\ \mathrm{V})$, update $\tilde{\mathrm{V}} = \mathrm{V}$;
9: **end while**
10: according to $\Theta = \mathrm{diag}\left(u_{\max}(\mathrm{V})\sqrt{\lambda_{\max}(\mathrm{V})}\right)[1:N]$, obtain the IRS coefficient matrix Θ.

4. Simulation And Result Analysis

In order to illustrate the channel gain brought by the active IRS, compared with the passive IRS, the fading model of each channel adopted in this section is the same as that in Fig. 2. Using the parameters in Fig. 2, we simulate and analyze the safety performance of the active IRS-assisted communication system. Where "active-IRS" represents the simulation curve in the presence of active IRS, "without-IRS" represents the simulation curve without IRS, and "passive-IRS" represents the simulation curve in the presence of passive IRS. In Fig. 2, the transmit power of the base station is set as $p_t = 40dBm$, and we compared the curve of the relationship between the number of reflection units N and the safety rate of the IRS under different schemes. The interference power threshold of the active IRS is $\overline{P}_A = 30dBm$, and the maximum value of the amplitude amplification is set as $a^2_{\max} = 35dB$. As the number of reflector elements increases, the security rate of both passive IRS and active IRS-aided communication systems increases. Under the same parameter settings, the performance gain brought by the active IRS is significantly better than that of the passive IRS and the communication system without IRS. As can be seen from the figure, due to the influence of "double fading", the number of reflection units required by the active IRS is smaller than the number of reflection units required by the passive IRS when the same safe rate is achieved. This shows that under the condition of achieving the same performance gain, the active IRS can save the number of reflection units and reduce the complexity of the IRS. When the legitimate users are far away from the IRS, that is $x_{IRS} = 40m$, the number of IRS reflection units has little effect on the system performance. The signal received by the user is mainly the signal reflected by the

IRS, and the average safe rate increases significantly with the increase of the number of reflecting units.

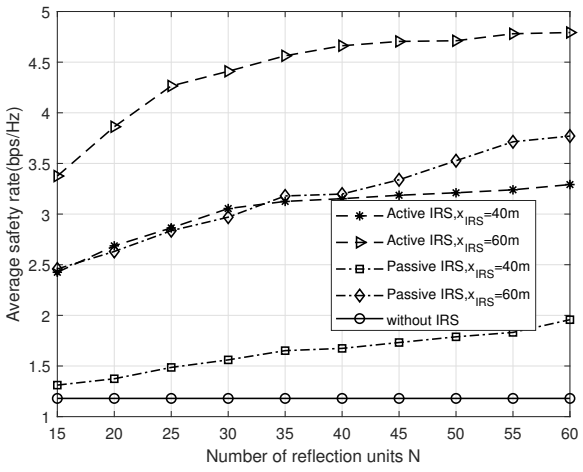


Fig. 2. Average safety rate of different reflection units

In Fig. 3, the graph is the relationship between the transmission power of different base stations and the system security rate. We set the number of reflection units $N = 32$ of IRS, and compare the improvement of system safety performance between active IRS and passive IRS under the same parameter settings. The interference power threshold of IRS is fixed as $\overline{P}_A = 20dBm$, and the maximum value of the amplitude amplification is set as $a^2_{\max} = 35dB$. It can be seen from the figure that the larger the value of p_t, the higher the security rate of both passive IRS and active IRS-assisted communication systems. But when the power increases to a certain value, the upward trend of the safe rate curve using the active IRS scheme is gradually smoothed. This is because it is subject to the maximum interference power constraint of the IRS. Therefore, the achievable performance gain offered by active IRS is limited. Compared with communication systems without IRS, IRS can significantly improve the security performance of the system. This is because through the proposed optimization algorithm, the performance loss of the signal reflected by the IRS received by the legitimate user is smaller than that of the eavesdropping user, thereby improving the security rate of the system.

In Fig. 4, the comparative active IRS and passive IRS system safety rate curves under different parameter settings. The horizontal coordinate of the IRS is set as $x_{IRS} = 60m$, the transmit power of the base station is $p_t = 40dBm$, and the interference power threshold of the IRS is $\overline{P}_A = 20dBm$. As can be seen from the figure, the achievable performance gain provided by the active IRS is significantly better than that of the passive IRS. With the increase of $a^2_{\max}$, the security rate of the active IRS-assisted communication system is continuously improved. This shows that the active IRS can enhance the incident signal by increasing the amplification factor and the number of reflective elements.

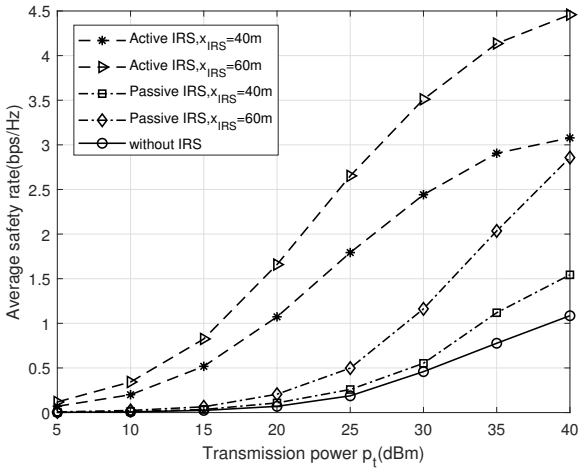


Fig. 3. Average safe rate for different transmit powers p_t

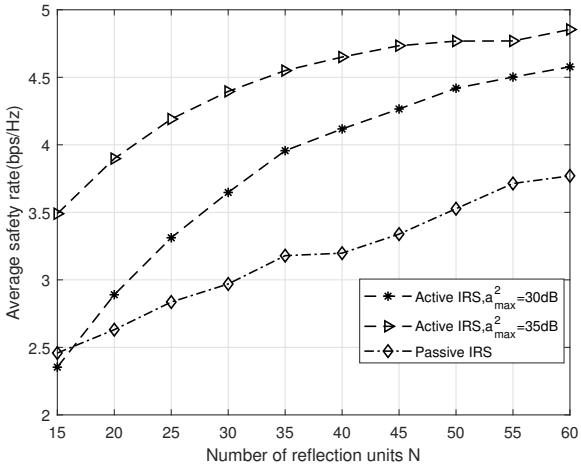


Fig. 4. Average safe rate for different amplification coefficients $a^2_{\max}$

5. Conclusion

In this chapter, we studied the IRS-assisted single-input single-output secure communication system, designed the optimal phase shift based on two types of IRS, passive and active, and proposed a secure rate maximization algorithm for IRS-assisted wireless communication networks. For the passive IRS-assisted communication system, the influence of the number of IRS reflective elements on the security performance of the system is studied. The simulation results show that the more reflective units in the IRS, the higher the performance gain brought by the IRS. However, due to the influence of "double fading", passive IRS improves the performance of the system to a limited extent. When the number of reflection elements increases to a certain value, the slope of the average safe rate curve also decreases continuously. For the communication system assisted by the active intelligent reflective surface, the active IRS can not only adjust the phase shift of the signal, but also amplify the amplitude of the signal. In order to solve the non-convex secrecy rate optimization problem based on this design, a min-max optimization algorithm is proposed. The simulation results show that the influence of "double fading" effect can be effectively alleviated with the assistance of active IRS. Under the actual power consumption model, the performance gain brought by the proposed active IRS-assisted system is better than that of the traditional passive IRS system.

Acknowledgments. This work was supported by the Open Fund of Advanced Cryptography and System Security Key Laboratory of Sichuan Province under Grant SKLACSS-202115, the Projects of Hunan Provincial Department of Education Department under Grant 21A0408 and 20C0879, and the Hunan Province for Postgraduate No.QL20220248.

References

1. Basar, E., Di Renzo, M., De Rosny, J., Debbah, M., Alouini, M.S., Zhang, R.: Wireless communications through reconfigurable intelligent surfaces. IEEE access 7, 116753–116773 (2019)
2. Lavigne, G., Achouri, K., Asadchy, V.S., Tretyakov, S.A., Caloz, C.: Susceptibility derivation and experimental demonstration of refracting metasurfaces without spurious diffraction. IEEE Transactions on Antennas and Propagation 66(3), 1321–1330 (2018)
3. Liu, F., Pitilakis, A., Mirmoosa, M.S., Tsilipakos, O., Wang, X., Tasolamprou, A.C., Abadal, S., Cabellos-Aparicio, A., Alarcón, E., Liaskos, C., et al.: Programmable metasurfaces: State of the art and prospects. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 1–5. IEEE (2018)
4. ElMossallamy, M.A., Zhang, H., Song, L., Seddik, K.G., Han, Z., Li, G.Y.: Reconfigurable intelligent surfaces for wireless communications: Principles, challenges, and opportunities. IEEE Transactions on Cognitive Communications and Networking 6(3), 990–1002 (2020)
5. Zhu, Z., Wang, Z., Xu, J., Wang, Z., Wang, N., Hao, W.: Future wireless communications assisted by smart reflectors: current status and prospects. Acta Aeronautica et Astronautica Sinica 43, 203–217 (2022)
6. Zhao, W., Wang, G., Atapattu, S., Tsiftsis, T.A., Tellambura, C.: Is backscatter link stronger than direct link in reconfigurable intelligent surface-assisted system? IEEE Communications Letters 24(6), 1342–1346 (2020)
7. Huang, C., Zappone, A., Alexandropoulos, G.C., Debbah, M., Yuen, C.: Reconfigurable intelligent surfaces for energy efficiency in wireless communication. IEEE Transactions on Wireless Communications 18(8), 4157–4170 (2019)

8. Wang, P., Fang, J., Yuan, X., Chen, Z., Li, H.: Intelligent reflecting surface-assisted millimeter wave communications: Joint active and passive precoding design. IEEE Transactions on Vehicular Technology 69(12), 14960–14973 (2020)
9. Venkatesh, S., Lu, X., Saeidi, H., Sengupta, K.: A high-speed programmable and scalable terahertz holographic metasurface based on tiled cmos chips. Nature electronics 3(12), 785–793 (2020)
10. Zhang, Z., Dai, L., Chen, X., Liu, C., Yang, F., Schober, R., Poor, H.V.: Active ris vs. passive ris: Which will prevail in 6g? ArXiv abs/2103.15154 (2021)
11. Long, R., Liang, Y.C., Pei, Y., Larsson, E.G.: Active reconfigurable intelligent surface-aided wireless communications. IEEE Transactions on Wireless Communications 20(8), 4962–4975 (2021)
12. Zhi, K., Pan, C., Ren, H., Chai, K.K., Elkashlan, M.: Active ris versus passive ris: Which is superior with the same power budget? IEEE Communications Letters 26, 1150–1154 (2022)
13. Khoshafa, M.H., Ngatched, T.M.N., Ahmed, M.H., Ndjiongue, A.R.: Active reconfigurable intelligent surfaces-aided wireless communication system. IEEE Communications Letters 25(11), 3699–3703 (2021)
14. Zeng, P., Qiao, D., Wu, Q., Wu, Y.: Throughput maximization for active intelligent reflecting surface-aided wireless powered communications. IEEE Wireless Communications Letters 11, 992–996 (2022)
15. Xu, D., Yu, X., Ng, D.W.K., Schober, R.: Resource allocation for active irs-assisted multiuser communication systems. 2021 55th Asilomar Conference on Signals, Systems, and Computers pp. 113–119 (2021)
16. You, C., Zhang, R.: Wireless communication aided by intelligent reflecting surface: Active or passive? IEEE Wireless Communications Letters 10, 2659–2663 (2021)
17. Liu, K., Zhang, Z., Dai, L., Xu, S., Yang, F.: Active reconfigurable intelligent surface: Fully-connected or sub-connected? IEEE Communications Letters 26(1), 167–171 (2022)
18. Dai, L., Wang, B., Wang, M., Yang, X., Tan, J., Bi, S., Xu, S., Yang, F., Chen, Z., Renzo, M.D., Chae, C.B., Hanzo, L.: Reconfigurable intelligent surface-based wireless communications: Antenna design, prototyping, and experimental results. IEEE Access 8, 45913–45923 (2020)
19. Li, Q., Yang, L.: Beamforming for cooperative secure transmission in cognitive two-way relay networks. IEEE Transactions on Information Forensics and Security 15, 130–143 (2020)
20. Barros, J., D. Rodrigues, M.R.: Secrecy capacity of wireless channels. In: 2006 IEEE International Symposium on Information Theory. pp. 356–360 (2006)
21. Chen, J., Yang, L., Alouini, M.S.: Physical layer security for cooperative noma systems. IEEE Transactions on Vehicular Technology 67(5), 4645–4649 (2018)
22. Mucchi, L., Ronga, L., Zhou, X., Huang, K., Chen, Y., Wang, R.: A new metric for measuring the security of an environment: The secrecy pressure. IEEE Transactions on Wireless Communications 16(5), 3416–3430 (2017)
23. Cui, M., Zhang, G., Zhang, R.: Secure wireless communication via intelligent reflecting surface. IEEE Wireless Communications Letters 8(5), 1410–1414 (2019)
24. Yu, X., Xu, D., Schober, R.: Enabling secure wireless communications via intelligent reflecting surfaces. In: 2019 IEEE Global Communications Conference (GLOBECOM). pp. 1–6 (2019)
25. Wang, H., Bai, J., Dong, L.: Intelligent reflecting surfaces assisted secure transmission without eavesdropper's csi. IEEE Signal Processing Letters 27, 1300–1304 (2020)
26. Du, L., Huang, C., Guo, W., Ma, J., Ma, X., Tang, Y.: Reconfigurable intelligent surfaces assisted secure multicast communications. IEEE Wireless Communications Letters 9(10), 1673–1676 (2020)
27. Feng, B., Wu, Y., Zheng, M.: Secure transmission strategy for intelligent reflecting surface enhanced wireless system. 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP) pp. 1–6 (2019)
28. Chen, J., Liang, Y.C., Pei, Y., Guo, H.: Intelligent reflecting surface: A programmable wireless environment for physical layer security. IEEE Access 7, 82599–82612 (2019)

29. Dong, L., Wang, H.M., Bai, J.: Active reconfigurable intelligent surface aided secure transmission. IEEE Transactions on Vehicular Technology 71(2), 2181–2186 (2022)
30. Cheng, R., Yu, W., Song, Y., Chen, D., Ma, X., Cheng, Y.: Intelligent safe driving methods based on hybrid automata and ensemble cart algorithms for multihigh-speed trains. IEEE Transactions on Cybernetics 49(10), 3816–3826 (2019)
31. Cheng, Y., Jiang, H., Wang, F., Hua, Y., Feng, D., Guo, W., Wu, Y.: Using high-bandwidth networks efficiently for fast graph computation. IEEE Transactions on Parallel and Distributed Systems 30(5), 1170–1183 (2019)
32. Liu, G., Zhang, X., Guo, W., Huang, X., Liu, W.H., Chao, K.Y., Wang, T.C.: Timing-aware layer assignment for advanced process technologies considering via pillars. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 41(6), 1957–1970 (2022)
33. Hongju Cheng, Leihuo Wu, F.H.C.T., Yu, Z.: Data recovery in wireless sensor networks based on attribute correlation and extremely randomized trees. Journal of Ambient Intelligence and Humanized Computing pp. 245–259 (2021)
34. Dai, Y., Wang, S., Chen, X., Xu, C., Guo, W.: Generative adversarial networks based on wasserstein distance for knowledge graph embeddings. Knowledge-Based Systems 190, 105165 (2020)
35. Fu, Y.G., Huang, H.Y., Guan, Y., Wang, Y.M., Liu, W., Fang, W.J.: Ebrb cascade classifier for imbalanced data via rule weight updating. Knowledge-Based Systems 223, 107010 (2021)
36. Fu, Y.G., Zhuang, J.H., Chen, Y.P., Guo, L.K., Wang, Y.M.: A framework for optimizing extended belief rule base systems with improved ball trees. Knowledge-Based Systems 210, 106484 (2020)
37. Zhang, H., Li, J.L., Liu, X.M., Dong, C.: Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection. Future Generation Computer Systems 122, 130–143 (2021)

Yiming Li received the B.S degree in computer science and technology from the Computer Department of Hunan University in 2001,M.S degree in computer application technology from the school of Computer Science of Huazhong University of Science and Technology in 2006,she is currently a lecturer with the school of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang, China. Her research interests include communication algorithm, such as channel estimation, equalizer, encoding, decoding, and so on.

Xitao Liang received the B.S. degree in communication engineering from Guangdong Ocean University, Zhanjiang, China, in 2016.He is currently studying for a master's degree in Hunan Institute of Science and Technology. His research interests include convex optimization in wireless communication.

Wenwu Xie received the B.S., M.S., and Ph.D. degrees in communication engineering from Huazhong Normal University, Wuhan, China, in 2004, 2007, and 2017, respectively. He is currently an Associate Professor with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang, China. His research interests include communication algorithm, such as channel estimation, equalizer, encoding/decoding, etc.

Juan Zhu received M.S. degree of Pattern Recognition and Intelligent System from Kunming University of Science and Technology in 2011.She is currently a Lecturer in Hubei

University of Arts and Science. Her research interests include image processing technology and research on detection and recognition algorithm.

Received: January 31, 2020; Accepted: July 15, 2020.

Computer Science and Information Systems 20(2):609–629 https://doi.org/10.2298/CSIS220705012W

Content-only attention Network for Social Recommendation

Bin Wu[1], Tao Zhang[2] and Yeh-Cheng Chen[3]

[1] School of Internet of Things Engineering, Jiangnan University,
Wuxi, 214122, China
wubin@jiangnan.edu.cn
[2] China Ship Scientific Research Center,
Wuxi 214122, China
taozhang@jiangnan.edu.cn
[3] Department of computer science, University of California,
Davis, CA, USA
ycch@ucdavis.edu

Abstract. With the rapid growth of social Internet technology, social recommender has emerged as a major research hotspot in the recommendation systems. However, traditional graph neural networks does not consider the impact of noise generated by long-distance social relations on recommendation performance. In this work, a content-only multi-relational attention network (CMAN) is proposed for social recommendation. The proposed model owns the following advantages: (i) the comprehensive trust based on the historical interaction records of users and items are integrated into the recursive social dynamic modeling to obtain the comprehensive trust of different users; (ii) social trust information is captured based on the attention network mechanism, so as to solve the problem of weight distribution in the same level domain; (iii) two levels of attention mechanisms are merged into a unified framework to enhance each other. Experiments conducted on two representative datasets demonstrate that the proposed algorithm outperforms previous methods substantially.

Keywords: recommender system, social network, content-only multi-relational attention network.

1. Introduction

With the rapid development of computer technology and the maturity of Internet economy, recommendation systems have become a hot topic for researchers. The traditional recommendation systems mainly face two problems: one is the sparsity problem with the users' rating data. In practical applications, the number of users and items is vast, but the historical behavioral records between users and items are rare. When the number of users and items in the recommendation system increases, with the extremely sparse historical behavioral matrix, the user preferences cannot be accurately learned. Therefore, the accuracy of recommendation is significantly reduced. The other problem is the cold start problem. For users or items newly added to the

recommendation system, there is no historical behavioral record, so it is difficult to provide fine personalized recommendations.

How to get effective information from massive amounts of data is a huge challenge for ordinary users. Researchers have proposed to use the traditional matrix decomposition methods to improve the neural network recommendation models in recent years. For example, some works use deep learning schemes to model the deeper data relationship between the user eigenvector and the item eigenvector, which proactively provides them with products that meet their potential preferences [1]. In addition, the graph-based neural network is used to depict the user preference generation process, and the recommendation system can help them quickly find satisfactory information in large amounts of information, and for merchants, it can not only help to decide to push to specific users, but also can increase user loyalty through more satisfactory services [2, 3]. Although the recommendation algorithm in various applications have achieved great success, however, sparse data problem is still one of the important bottlenecks affecting algorithm performance recommendation algorithm is usually based on the user's historical data to model user preferences.

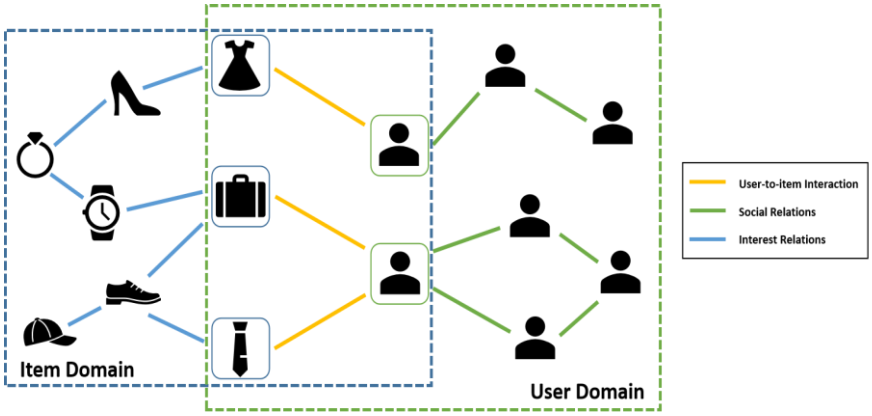


Fig. 1. The graph structure in social recommendation, which includes three graphs: the user-user graph (left part), the user-item graph (middle part) and the item-item graph (right part).

This paper mainly concern the recommendation algorithms combined with social information. In order to improve the quality of recommendation algorithms, especially in dealing with the problem of sparse data, a content-only multi-relational attention network (CMAN) is proposed, which jointly models the three graph structures for social recommendation, including the user-user graph, the item-item graph and the user-item graph (Fig. 1.). The contribution points of this paper mainly include:

First, a novel framework called Content-based Multi-Relational Attention Network (CMAN) is proposed for social recommendation, which jointly captures the influence and interest diffusion in multi-relational context space;

Second, this paper try to add homogeneous information between items to solve the data sparsity problem and high-order influence diffusion process is further exploited to extract multi-relational contexts;

Third, a two-level attention mechanism is proposed to comprehensively consider the influence of score similarity and implicit trust relationship on the trust between each group of users, and obtain a more accurate trust beard and recommendation model.

The remainder of this paper is organized as follows. In Section 2, the social recommendation problem is defined to be solved and review some related works. The proposed framework is formally described in Section 3. Experimental results and discussion are presented in Section 4. Finally, conclusion and future research directions are given in Section 5.

2. Problem Definition and Related Work

This part mainly introduces two classical algorithms directly related to this paper. First, the problem of social recommendation is formally defined and described, and then two existing related work are reviewed respectively, including the recommendation algorithm based on probability matrix decomposition.

Table 1. Variables used of this paper

Variables	Definitions
p_a	Latent embedding of user a
q_i	Latent embedding of item i
x_a	Real-valued attributes of a
y_i	Real-valued attributes of i
d	Length of the embedding vector
S	User-user social network
S_a	The set of social friends that a follows
R	User-item interaction matrix
$R_U(i)$	The set of users that interacted with item i
$R_I(a)$	The set of items that user a interacts with
F	Item-item influence network
F_i	The set of item friends that item i connects
τ	A fixed threshold who links both items in F
r_{ai}	The observed preference of user a in item i
$\hat{r}_{ai}$	The predicted preference of user a in item i
$\oplus$	Concatenation operator
G_S	User-user social graph
G_I	User-item interest graph
G_F	Item-item influence graph

2.1. Problem Definition

This paper assumes that $U = \{u_1, u_2, \mathrm{K}, u_n\}$ denotes the sets of users, and $V = \{v_1, v_2, \mathrm{K}, v_m\}$ is the sets of items, where *N* and *M* are numbers of users and items. A user-item interaction matrix $R \in ¡^{n \times m}$ is used to depict users' implicit preference and interests to items (In proposed model, rating values range from 1 to 5). This paper assume that $r_{ai} = 1$ if u_a is interested in v_i, and $r_{ai} = 0$ if u_a do not rate v_i. In addition, this paper uses $R_U(i)$ and $R_I(u)$ *K*-dimensional potential vector of user u and product, respectively. Moreover, the user-user directed graph is denoted as $G = [U, S \in ¡^{n*n}]$, where U denotes the set of users *u* and *S* is the connections relationship between users of a social network. This paper denotes $s_{ab} = 1$ if u_a trusts u_b and zero otherwise. In addition, this paper uses S_a to denote the set of users, the purpose of the probability matrix factorization is to learn these vectors from the score, i.e., $S_a = [b \mid s_{ab} = 1]$. Moreover, this paper introduces an embedding vector $x_a \in ¡^{d}$ to predict how the user *u* scores on the unrated items for u_a and an embedding vector $y_i \in ¡^{d}$ to predict how the user *v* scores on the unrated items for v_i, where d denotes the dimension of embedding vector, given the evaluation matrix R and the trust relationship T, it can be shown that recommendation algorithms combined with social information can effectively alleviate the problem of sparse data. The used notations of this paper are summarized in Table 1. The social recommendation problem can be described as:

Input: a user set *U*, an item set *V*, the user-item interaction matrix R, the user social network S and the real-valued attribute matrices X and Y of *U* and *V*.

Output: $\hat{R} = f(U, V, R, S, X, Y)$, where $\hat{R} \in ¡^{n*m}$ denotes the unobserved interactions between users and items.

2.2. Related Work

Classical CF Recommendation Models. There are two main types of collaborative filtering methods [4], i.e., 1) memory-based collaborative filtering, which compute the similarity between users and items through users' rating history, and then new items are recommended for users based on the similarity. Typical examples of this approach are neighborhood-based CF and item-based/user-based top-N recommendations [5, 6]; and 2) model-based collaborative filtering models, which are developed to predict users' rating of unrated items [7]. The focus of classical CF recommendation models is on how to integrate social information and evaluation information more effectively.

Most social recommendation algorithms focus on solving two problems: how to effectively integrate social information into recommendation algorithms? How to estimate the trust between users to improve the algorithm accuracy? For the first

problem, many work attempts to model social information from different perspectives [8]. For example, Jamah has proposed a TrustWalker algorithm for random walk in scoring networks and social networks. The algorithm obtains the results by running midstream in the historical data without pretraining, but when the data volume is relatively large, the inquiry time is often too large [9]. Assuming that the user-item rating matrix is $R \in ¡^{n \times m}$ (The purpose of the probability matrix factorization is to learn these vectors from the score), the matrix factorization algorithm usually learns two low-rank matrices $U \in ¡^{n*k}$ and $V \in ¡^{m*k}$, therefore, it can be formulated as:

$$R \approx \hat{R} = UV^T \quad (1)$$

where $\hat{R}$ denotes the approximation matrix of R, U denotes the user's latent feature matrix, and V represents the item's latent feature matrix. Generally speaking, the rank k of two characteristic matrices U and V is very small, so the above matrix factorization is also called low rank matrix factorization. After achieving U and V, the user a that corresponds to item i can be predicted according to the following criterion [10-12]:

$$\hat{r}_{ai} = U_a V_i^T \quad (2)$$

where u_a is the *a*-th row of the user embedding matrix *U*, which is a normal distribution as the mean. Similarly, v_i denotes the latent embedding of item i in i-th row of item embedding matrix *V*. To get the optimal matrix representation $U \in ¡^{n*k}$ and $V \in ¡^{m*k}$, additional L2-norm regularization terms [8] are used to solve this problem:

$$L = \sum_{a=1}^{n} \sum_{i=1}^{m} \left(r_{ai} - u_a v_i^T \right)^2 + \mu \| U \|_F^2 + \mu \| V \|_F^2 \quad (3)$$

where the first term is the approximation error of matrix decomposition, the second and third terms are regularization terms, and μ is the regularization coefficient. Higher accuracy was achieved with their algorithm compared to previous work.

While model-based recommendation models significantly reduce the memory requirement and computation complexity, SVD [13], matrix factorization (MF) [14, 15] and non-negative matrix factorization (NFM) [16] are widely used recommendation methods, and the implied similarity of the trust relationship is considered.

Matrix Factorization-Based Social Recommendation Models. Social-based recommendation has gradually become an indispensable part of recommendation algorithms. The focus of social recommendation algorithm is on how to integrate social information and evaluation information more effectively [17-19]. Traditional recommendation systems assume that users are independent and identically distributed, which subconsciously ignores the social interaction between users. However, these algorithms always face the problem of very sparse user history data, resulting in decreased recommendation quality. To solve this problem, it is effective to introduce auxiliary data or mine more laws in the data into the algorithm.

The first category is based on the shared representation of the user feature matrix, which means that the dynamic combination with the user and product similarity to improve the algorithm to process sparse data. By assuming the user evaluation is

determined by personal preferences and friend influence, the objective function of SoRec [19] can be described as:

$$F_{SoRec} = \sum_{r_{ai} \neq 0} \left(r_{ai} - g\left(u_a v_i^T\right)\right)^2 + \lambda_u \sum_{S_{ab \neq 0}} \left(S_{ab} - g\left(u_a z_b^T\right)\right)^2 + \lambda_r \left(\| U \|_F^2 + \| V \|_F^2 + \| Z \|_F^2\right) \quad (4)$$

where $g(x) = \frac{1}{1+\exp(-x)}$, $z_b \in ¡^{d}$ denotes the social attribute representation of user *b*, which is the b-th row of the social attribute matrix $Z \in ¡^{n*d}$ and $u_a z_b^T$ denotes the predicted social relationship between user a and user b, which is fitted by the user feature vector u_a and social feature vector z_b. Different from SoRec, TrustMF [20] is the SoRec algorithm, assuming that users share the same preference vector in the evaluation network and social network, and using the probability matrix decomposition to finally obtain the recommendation structure considering the user's friend factors. Higher accuracy was achieved with their algorithm compared to previous work. Followed it, Fang et al. [21] proposed RSTE, a recommendation algorithm that integrates evaluation information and social networks. In addition, Tang et al. [22] proposed a social recommendation model LOCABAL, The algorithm assumes that user evaluation is determined by personal preferences and friend influence, and has a balance.

Guo et al. [23] introduced the method of trust communication into the recommendation algorithm (SocialMF) [24]. This model spreads the trust relationship by restricting the average preferences of users and their friends that are similar, so as to get more accurate results, therefore, our objective function could be rewritten as:

$$\sum_{r_{ai} \neq 0} \left(r_{ai} - \mu - b_a - b_i - \left(u_a + |R(a)|^{-\frac{1}{2}} \sum_{p \in R(a)} I_p + |S(a)|^{-\frac{1}{2}} \sum_{k \in S(a)} W_k \right)^T v_i \right)^2 \quad (5)$$

where μ denotes the mean value of global ratings, b_a denotes the difference between the average rating of u_a and μ, b_i denotes the difference between the average rating of v_i and μ, I_p indicates the influence factor of v_p on user feature vector, W_k indicates the influence factor of u_k on user feature vector, and $R(a)$ denotes the collection of items rated by u_a, $S(a)$ denotes the social association user set of u_a.

Graph Neural Network-Based Social Recommendation Models. Graph Neural Networks (GNNs), as a generalization of deep neural networks on graph data[25], can better extract and represent data characteristics in graph field. GNN has developed a lot of different forms, such as GCN [26], GAT[27], GraphSAGE [28] and so on. In this regard, many works considered the impact of user score similarity on trust strength, using social networks as a regular constraint term to learn user preferences, and obtained more accurate results. Guo Lei et al. used the potential vectors obtained from the probability matrix decomposition of the scoring matrix to calculate the similarity between users and friends, which improves the accuracy of the algorithm. They also proposed to make full use of the relationship between objects to improve the recommendation accuracy.

The development of graph neural networks assumes that each friend has the same impact on the user. In recent years, the related research based on graph neural networks in recommender systems has attracted more and more attention of scholars. GC-MC [29]

considered the different cases of users as trusted people and trusted people, and calculates the trust similarity of the two cases respectively, so as to restrict the learning of user preference vectors, and then weight the user vectors in these two cases and influence the scoring results.

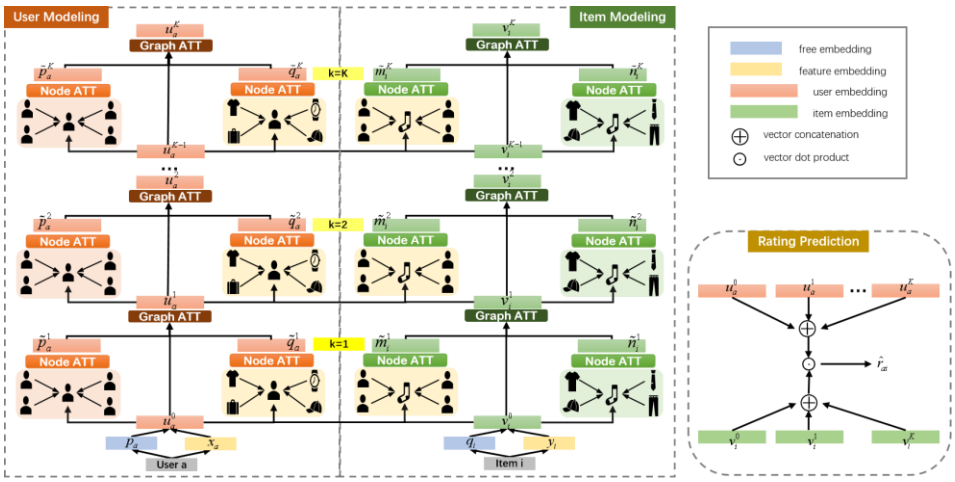


Fig. 2. The overall architecture of the proposed CMAN model.

Considered the trust strength between users from the perspective of scoring and trust data, the latent relationship (cooperative signal) is proposed to improve user's ability measure function and reliability measure function, respectively, and obtained the estimated users' preferences by combining them with the score similarity function. A more accurate recommendation model is also constructed. Similarly, PinSage [30] is a random-walk-based GCN that uses probabilistic matrix factorization to obtain the potential vectors of the user as trusted and trusted person, and use this to calculate the similarity of the user as trusted and trusted, respectively.

The above methods directly use GNNs over the user-item interaction data without considering cold start and sparse information. For comparison, GraphRec [31] proposed a unified framework for jointly modeling user/item, In fact, the trust relationship among users is influenced by a variety of factors, some by similar hobbies, some by the same social circle, and some just out of politeness. A simple binary trust network does not reflect the size of the influence between users, nor does it fully exploit the implied user preferences information in social networks, which leads to the second question, namely, how to estimate the trust between users [27]. Moreover, DiffNet [32] used the potential vectors obtained from the probability matrix decomposition of the scoring matrix to calculate the similarity between users and friends, which improves the accuracy of the algorithm. They also proposed to make full use of the relationship between objects to improve the recommendation accuracy. DiffNet++ [33] further established the relationship between user scoring preferences and social preferences through the linear mapping method to aggregate the different order neighbors' feature vectors for each channel.

This paper uses probabilistic matrix factorization to obtain the potential vectors of the user as trusted and trusted person, and uses this to calculate the similarity of the user as trusted and trusted, respectively. Besides that, this paper comprehensively considered the effect of score and trust similarity on the strength of trust among users.

3. The Proposed Model

In this section, we will first give an overview about the architecture of our proposed model CMAN, and then detail each component of the model. Finally, we give the training process of CMAN.

3.1. An Overview of the Proposed Model

The overall architecture of the proposed model is shown in **Fig. 2**. Generally speaking, the proposed CMAN model consists of three components, i.e., user modeling, item modeling, and rating prediction. At the beginning of user modeling and item modeling, we first integrate free embedding and feature embedding to get the initial low-dimension user/item representation. Compared to the previous work of social recommendation, our trust similarity calculation does not depend on the common friend collection or the number of user friends between the users. Stable and reliable calculation results are obtained even when the trust data is sparse or the set of common friends is zero. Moreover, multi-layer GNNs with two different attention mechanism captures the multi-order influence information at different scales. Meanwhile, by mapping user social behavior to low-dimensional subspaces, it shows that user vectors not only contain direct associations between users, but also imply indirect connections between users. Therefore, a more accurate trust-most model is obtained. Last, we comprehensively considered the effect of score and trust similarity on the strength of trust among users.

3.2. User Modeling

This paper defines $P \in ¡^{n*d}$ as the embedding matrices of users, where d is the embedding size and p_a denotes the free latent embedding for user a. By feeding p_a and the associated feature vector x_a into the fusion layer, the initial latent preference of user a is defined as:

$$u_a^0 = \sigma\left(\mathrm{W}_1 \times \left[\mathrm{p}_a, \mathrm{x}_a\right]\right) \quad (6)$$

where $\sigma(\cdot)$ denotes the activation function, W_1 denotes a trainable transformation matrix, based on this, the information in the scoring and trust data is also considered in close proximity to our work.

General GNN-based social recommendation methods leverage two different graphs, i.e., a user-user social graph G_S and a user-item interest graph G_I. Two aspects of information are combined, and the implied similarity of the trust relationship is considered, thus obtaining a more accurate trust. This paper defines p_a^k as the aggregated embedding of influence diffusion from the trusted social neighbors of G_S and q_a^k as the embedding of interest diffusion from the interested item neighbors of G_I at the k-th layer. Finally, we obtain a recommendation method model that comprehensively considers the scoring and trust similarity, which can be defined as:

$$u_a^k = \left(\gamma_{a1}^k u_a^{k-1} + \gamma_{a2}^k p_a^{\sim k} + \gamma_{a3}^k q_a^{\sim k} \right), \quad (7)$$

$$p_a^{\sim k} = \sum_{b \in S_a} \alpha_{ab}^k u_b^{k-1}, \quad (8)$$

$$q_a^{\sim k} = \sum_{i \in R_I(a)} \beta_{ai}^k v_i^{k-1}, \quad (9)$$

where u_a^{k-1} denotes the latent embedding of user a at the $(k-1)$-th, p_a^k denotes the user-based social influence diffusion process and q_a^k denotes the item-based interest influence diffusion process from two graphs respectively. In social networks, α_{ab}^k denotes the social influence of user b to a at the k-th layer in G_S, β_{ai}^k denotes the interest influence of item i to user a at the k-th layer in G_I, and γ_{al}^k denotes the graph level weight.

The user's potential preference information is usually implied in the scoring matrix, which is the main data adopted by the recommendation algorithm. Users' social relations usually can only obtain binary data, but not all friends have the same impact on users.The basic goal of the social recommendation algorithm is to predict how the user u scores on the unrated items *i*, given the evaluation matrix *R* and the trust relationship T. Specifically, the node-level weights, i.e., the social influence strengths α_{ab}^k and the interest influence strengths β_{ai}^k, concretely show that recommendation algorithms combined with social information can effectively alleviate the problem of sparse data. At present, how to efficiently mine the hidden user preference information in social relationships is the focus of social recommendation algorithm. Traditional collaborative filtering recommendation algorithms usually use users only user-history evaluation data to model them, and then predict users 'future evaluation and selection, including memory-based and model-based 3' women. Therefore, it is necessary for each user to build personalized weight. Thus, this step can quickly generate the recommended results, and cosine similarity functions α_{ab}^k are defined:

$$\alpha_{ab}^k = \frac{u_a^{k-1} \cdot u_b^{k-1}}{|u_a^{k-1}| \times |u_b^{k-1}|}, \quad (10)$$

$$\alpha_{ab}^{k} = soft\max\left(\alpha_{ab}^{k}\right) = \frac{\exp\left(\alpha_{ab}^{k}\right)}{\sum_{b\in S_a}\exp\left(\alpha_{ab}^{k}\right)}, \quad (11)$$

Where we use a softmax function and each value is quantified into (0,1). Similarly, we calculate the interest influence score β_{ai}^{k} by corresponding two vector product and item embedding as input. Then, the conditional probability of the scoring matrix R in the given item i is defined as:

$$\beta_{ai}^{k} = \sigma\left(u_a^{k-1} \odot v_i^{k-1}\right), \quad (12)$$

$$\beta_{ai}^{k} = soft\max\left(\beta_{ai}^{k}\right) = \frac{\exp\left(\beta_{ai}^{k}\right)}{\sum_{i\in R_l(a)}\exp\left(\beta_{ai}^{k}\right)}, \quad (13)$$

where σ is the sigmoid function. Meanwhile, in order to prevent overfitting, the algorithm assumes that the potential vectors of the user and the product satisfy the Gaussian distribution. Inspired by the scheme of tackling the node attention layer, we can model the graph attention weights of $\gamma_{al}^{k}(l=1,2,3)$ as:

$$\gamma_{a1}^{k} = MLP_1^{k}\left(u_a^{k-1}\right), \quad (14)$$

$$\gamma_{a2}^{k} = MLP_2^{k}([u_a^{k-1}, p_a^{\sim k}]), \quad (15)$$

$$\gamma_{a3}^{k} = MLP_3^{k}([u_a^{k-1}, q_a^{\sim k}]), \quad (16)$$

where MultiLayer Perceptrons (MLPs) approaches are used to learn the graph attention weights at the $(k-1)$-th layer (u_a^{k-1}) and node attention representations at the k-th layer(p_a^{k} and q_a^{k}). Suppose that U- -RKXN and V6RK money represent the potential matrix of user and product and product, respectively, where U " and K represent the K-dimensional potential vector of user U and product, respectively. They model both from the perspective of trusted and scored close neighbor sets, however, these two sets are not equal, using only one aspect of the information for close neighbors present in only one of the sets. In addition, considering $\gamma_{a1}^{k}+\gamma_{a2}^{k}+\gamma_{a3}^{k}=1$, if the value of γ_{a2}^{k} is bigger than that of γ_{a3}^{k}, the effect of influence diffusion is greater than that of interest diffusion, and larger $\gamma_{a2}^{k}+\gamma_{a3}^{k}$ denotes that user embedding at layer k will be more affected by the two influence diffusion effects.

3.3. Item Modeling

$Q \in ¡^{m*d}$ indicates the free embedding matrices of items, where d denotes the embedding size and q_i represents the free latent embedding for item *i*. By incorporating q_i and the associated feature vector y_i into the fusion layer, we can get the initial item embedding:

$$v_i^0 = \sigma(W_2 \times [q_i, y_i,]), \quad (17)$$

In this paper, we attempt to construct an item-item influence network F. In order to more effectively model the social relationship between users, and fully explore the influence of social interaction and friends on users' preferences, a lot of work has made a beneficial exploration of this [34, 35].

For any item i and item j, we define their similarity coefficients s_{ij} as the number of users who liked both items. This paper defines the item implicit network as the graph $G_F = [V, F \in ¡^{m*m}]$, where V denotes the set of items and F denotes the connections between the two related items.

Based on the assumption that users and friends have similar preferences, for each item i, what is needed is to aggregate user-space information from the set of users who have interacted with item i, denoted as $R_U(i)$, including the recommendation algorithm based on probability matrix decomposition with item i, denoted as F_i. For each item i, given its $(k-1)$-th layer embedding u_a^{k-1} and v_i^{k-1}, we model the updated item embedding v_i^k at the k-th layer as:

$$v_i^k = (\eta_{i1}^k v_i^{k-1} + \eta_{i2}^k m_i^{\sim k} + \eta_{i3}^k n_i^{\sim k}), \quad (18)$$

$$m_i^{\sim k} = \sum_{j \in F_i} \mu_{ij}^k v_j^{k-1}, \quad (19)$$

$$n_i^{\sim k} = \sum_{a \in R_U(i)} v_{ia}^k u_a^{k-1}, \quad (20)$$

where $R_U(i) = [a \mid r_{ia} = 1]$ denotes the user set that rates item i, $F_i = [j \mid f_{ij} = 1]$ denotes the item set that is related to item i, $m_i^{\sim k}$ is the item i's aggregated embedding in the item-item influence graph G_F, $n_i^{\sim k}$ represents the item i's aggregated embedding in the user-item interest graph G_I, and $\eta_{il}^k (l = 1, 2, 3)$ is the aggregation weight. Considering that the trust intensity of users to each friend is different, we calculate the interest attention weights $\tilde{m}_i^k$ between node i and its user node neighbors, and the influence attention weights $\tilde{m}_i^k$ between node i and its related node neighbors. The similarity can be measured from the perspective of scoring or trust data, given item's node representation v_i^{k-1} and all of its selected neighbors are described as:

$$\mu_{ij}^k = \frac{v_i^{k-1} \cdot v_j^{k-1}}{|v_i^{k-1}| \times |v_j^{k-1}|}, \quad (21)$$

$$v_{ia}^k = \sigma(v_i^{k-1} \odot u_a^{k-1}), \quad (22)$$

Because the graph attention weights $\gamma_{al}^{k}(l=1,2,3)$ in user-space is achieved, here an attention network is used to learn the item graph attention weight $\eta_{il}^{k}(l=1,2,3)$:

$$\eta_{i1}^{k} = MLP_4^k(v_i^{k-1}), \quad (23)$$

$$\eta_{i2}^{k} = MLP_5^k([v_i^{k-1}, m_i^{\sim k}]), \quad (24)$$

$$\eta_{i3}^{k} = MLP_6^k([v_i^{k-1}, n_i^{\sim k}]), \quad (25)$$

where other MLPs are used to learn the graph attention weights with the related item embedding at the $(k-1)$-th layer (v_i^{k-1}) and node attention representations at the k-th layer(m_i^{k} and n_i^{k}).

3.4. Rating Prediction

This paper comprehensively considered the effect of score and trust similarity on the strength of trust among users. With the latent embedding of user a and item i at layer k (i.e., u_a^k and v_i^k) for $k=[0,1,2,...,K]$, we can first concatenate them at each layer to get the final user embedding $u_a^* = [u_a^0 \| u_a^1 \| ... \| u_a^K]$ and the final item embedding $v_i^* = [v_i^0 \| v_i^1 \| ... \| v_i^K]$. Then, the user's potential preference information is usually implied in the scoring matrix, which is the main data adopted by the recommendation algorithm:

$$\hat{r}_{ai} = [u_a^0 \| u_a^1 ... \| u_a^K]^T [v_i^0 \| v_i^1 \| ... \| v_i^K], \quad (26)$$

3.5. Model Training

To give the parameters of CMAN, Bayesian Personalized Ranking loss (BPR) function[12] is used for training, the purpose of ranking task is to learn these vectors from the score[33, 36, 37]. The loss function is defined as:

$$L = \min_{\Theta} \sum_{(a,i^+,i^-)\in R} -\ln \sigma(\hat{r}_{ai^+} - \hat{r}_{ai^-}) + \lambda \| \Theta \|_2^2 \quad (27)$$

where $R = \{(a,i^+,i^-) | (a,i+) \in R^+, (a,i^-) \in R^-\}$ denotes the training set, R^+ represents the set of positive samples and R^- represents the set of negative samples (the user scoring matrix can learn the potential vectors of users and products to make predictions). $\sigma(x)$ is sigmoid function and Θ is regularization parameters set, i.e., $\Theta = [P, Q, W_1, W_2, [MLP_i^k]_{i=1,2,3,4,5,6}]$.

4. Experimental Results and Analysis

Yelp. Users in Yelp can rate local services and follow others that they like. The original dataset contains two parts of information, i.e., the directed interactive relationships among users, as well as the users' ratings to locations. There are five levels of ratings from 1 to 5 (Since scoring and trust data are usually sparse, the values of 0 and K shown in Table 2 are relatively small). Similar to previous works, this paper considers the ratings larger than 3 as "My Likes" of this user.

Flickr. Flickr is an online photo sharing website. Users follow other users and share funny images to friends, family and social media followers based on their preferences. The original dataset provides a great deal of preference information and social information.

4.1. Experiments Settings

This paper evaluated our proposed model on two representative data sets, Yelp and Flickr, from the authors of [32,33]. As they did in the study, this paper kept only users with at least two ratings and two social links, and filters with fewer than two interactions. In addition, this paper performs additional preprocessing steps to extract at least 2 similar item pairs that users like and use them as the edge information of our model. Note that item pairs (item links) are very sparse, so let's further consider the available links. The statistical results of the final data set are summarized in **Table 2**. This paper randomly selected 85% of the data for training, 5% for validation, and the remaining 10% for testing.

Table 2. The statistics of the two datasets

Dataset	Yelp	Flickr
# of Users	17,237	8,358
# of Items	38,342	82,120
# of Ratings	204,448	327,815
# of Density (Ratings)	0.03%	0.05%
# of Social Connections	143,765	187,273
# of Density (Social Relations)	0.05%	0.27%
# of Item Connections	79,876	498,664
# of Density (Item Relations)	0.011%	0.015%

To evaluate the top-K recommendation performance of the model, this paper used the recall based measure HR@K(hit rate) and the rank-based measure NDCG@K(normalized discount cumulative gain), which are widely used in top-K recommendation tasks [33,38]. Specifically, HR@K measures the percentage of test items that are successfully recommended in the top-K recommendation list, and NDCG@K further considers the ranking of test items in the top-K recommendation list. For these two indicators, the higher the value, the better the recommendation result. In our experiment, for many recommended tasks [32,39], this paper randomly selected 1000 unrated items for each user as negative items. We repeated each experiment 10 times and reported the average score of optimal performance for both indicators.

To evaluate the performance, this paper compare our CMAN against ten SOA baselines including traditional CF methods, social based recommender approaches and graph neural network based models. The baselines are detailed as below.

BPR [40]: A typical pair-wise algorithm that is derived from the maximum posterior estimator, only using the interaction data between users and items.

FM [10]: A powerful matrix decomposition method which considers pairwise feature interactions.

SocialMF [41]: A matrix factorization technique with trust propagation for recommendation in social networks.

TrustSVD [24]: A social recommendation method that incorporates first order social relations into modeling process.

ContextMF [42]: A fast and context-aware embedding learning method for social recommendation.

GraphRec [31]: A network embedding approach that employs attention mechanism to encode social network.

PinSage [30]: A random-walk Graph Convolutional Network that is highly-scalable and capable of learning embeddings for nodes in web-scale graphs containing billions of objects.

NGCF [2]: A deep neural network based framework leveraging high-order signals in user-item bipartite graph.

DiffNet[32]: A graph neural network based model that simulates social influence propagation.

DiffNet++ [33]: A Neural Influence and Interest Diffusion Network for social recommendation.

LCELS [26]:A low-dimensional space Diffusion Network for social recommendation.

Table 3. Comparison of the baselines

Model	Model Input		Model Embedding Ability			
	F	S	UU	UI	IU	II
BPR[40]	×	×	×	√	×	×
FM[10]	√	×	×	√	×	×
SocialMF[41]	×	√	√	√	×	×
TrustSVD[24]	×	√	√	√	×	×
ContextMF[42]	√	√	√	√	×	×
GraphRec[31]	×	√	√	√	×	×
PinSage[30]	√	×	×	√	√	×
NGCF[2]	×	×	×	√	√	×
DiffNet[32]	√	√	√	√	×	×
DiffNet++[33]	√	√	√	√	√	×
CMAN	√	√	√	√	√	√
CMAN-nf	×	√	√	√	√	√
CMAN-ns	√	×	×	√	√	√
CMAN-nii	√	√	√	√	√	×

Table 3 shows all the baselines and the key features of our models, showing what information each model leverages. Specifically, we use "F" for feature input and "S" for social network input. In the modeling process, "UU" and "UI" were used to represent

social information and interest information used for user embedded learning, and "IU" and "II" were used to represent interest information and project homogeneity information used for project embedded learning. Note that our proposed CMAN is the only one of these models that considers project homogeneity information. Because our proposed model, CMAN, is flexible and can be reduced to a simpler version, we also constructed several variants of CMAN as ablation studies. We use CMAN-NF, cman-NS, and Cman-NII to represent a simplified version of CMAN when deleting user and item characteristics, deleting social network input, and deleting item homogeneity information.

We implemented our proposed model using the Tensorflow framework, which optimized all models using the Adam optimizer with a batch size of 512. ,32,64 [16] and [0.0005, 0.001, 0.005, 0.01, 0.05, 0.1] search is embedded in size and more. We randomly initialize the user/item free embedding parameters and weight parameters with a Gaussian distribution, where the mean and standard deviation of all models are set to 0 and 0.1, respectively. SRAN in our proposed model, we in [0.0001, 0.0003, 0.001, 0.003, 0.01] in search of regularization parameter, and the proposed model reaches the best performance for Yelp dataset and Filckr dataset respectively when $\lambda_1 = 0.001, \lambda_2 = 0.003$. In addition, we have empirically set the hidden layer size to be the same as the embedded size and the activation function to be the same as Leaky ReLU. We carefully adjust the parameters of all baselines to ensure optimal performance for fair comparison.

4.2. Performance of Our Model and Baselines (RQ1)

We start by comparing CMAN's Top-10 recommendation performance with other baselines. Table 3 presents the overall rating prediction accuracy w.r.t. HR and NDCG for the recommendation methods with different embedding sizes *D* on the Yelp and Flickr data sets. We observe the following points. Firstly, the model based on graph neural network usually has better performance than the traditional model, including the classic CF model (such as BPR [12], FM[10]) and the society-based recommendation method (such as SocialMF [41], TrustSVD[24], ContextMF[42]). This observation makes sense because traditional models fail to capture important nonlinear relationships between users and objects. However, graph neural network-based models take into account higher-order social networks or higher-order user-item interaction information. The second observation is that, compared with other methods (such as PinSage[30], NGCF[2]), models with attention mechanism (such as GraphRec[31], diffnet++[33]) obtain better performance. This is not surprising, as this attention mechanism helps to better understand the implicit relationships between different nodes and aspects, improving recommendation performance. Thirdly, both social information and interest information play an important role in improving recommendation results. The performance of diffnet++ was significantly better than other baselines, making it the strongest baseline model, and our SRAN provided the best performance across all data sets (shown in Table 4).

Table 4. Overall comparison of HR@10 and NDCG@10 with different dimension size D

Model	Yelp				Flickr			
	HR		NDCG		HR		NDCG	
	D=32	D=64	D=32	D=64	D=32	D=64	D=16	D=64
BPR[12]	0.261	0.263	0.157	0.155	0.081	0.079	0.061	0.063
FM[10]	0.283	0.286	0.172	0.172	0.121	0.123	0.087	0.095
SocialMF [41]	0.271	0.279	0.169	0.168	0.106	0.117	0.086	0.096
TrustSVD [24]	0.285	0.294	0.171	0.175	0.134	0.140	0.106	0.108
ContextMF [42]	0.301	0.304	0.181	0.182	0.138	0.143	0.109	0.110
GraphRec [31]	0.291	0.291	0.168	0.181	0.121	0.123	0.090	0.093
PinSage[30]	0.296	0.305	0.179	0.186	0.123	0.126	0.094	0.099
NGCF[2]	0.307	0.304	0.184	0.188	0.115	0.119	0.088	0.094
DiffNet[32]	0.344	0.346	0.209	0.212	0.159	0.166	0.112	0.127
DiffNet++ [33]	0.355	0.369	0.216	0.226	0.168	0.183	0.121	0.142
LCELS [26]	0.339	0.342	0.199	0.209	0.147	0.151	0.107	0.115
CMAN	**0.363**	**0.384**	**0.223**	**0.238**	**0.175**	**0.197**	**0.127**	**0.154**
CMAN-ns	0.357	0.376	0.223	0.234	0.171	0.189	0.121	0.148
CMAN-nii	0.361	0.380	0.222	0.237	0.173	0.195	0.126	0.152

Table 5. Overall comparison of HR@N and NDCG@N with different top-N values (D=64)

Model	Yelp				Flickr			
	HR		NDCG		HR		NDCG	
	N=10	N=15	N=10	N=15	N=10	N=15	N=10	N=15
BPR[12]	0.263	0.325	0.155	0.175	0.079	0.103	0.062	0.073
FM[10]	0.282	0.344	0.171	0.187	0.123	0.147	0.095	0.106
SocialMF [41]	0.278	0.336	0.167	0.184	0.117	0.130	0.096	0.106
TrustSVD [24]	0.293	0.369	0.174	0.198	0.140	0.173	0.108	0.120
ContextMF [42]	0.304	0.383	0.182	0.208	0.143	0.176	0.110	0.113
GraphRec [31]	0.291	0.362	0.181	0.195	0.123	0.148	0.093	0.099
PinSage[30]	0.305	0.386	0.185	0.214	0.126	0.150	0.099	0.105
NGCF[2]	0.304	0.375	0.183	0.204	0.119	0.139	0.094	0.099
DiffNet[32]	0.346	0.422	0.212	0.231	0.166	0.185	0.127	0.130
DiffNet++ [33]	0.369	0.449	0.226	0.249	0.183	0.220	0.142	0.154
LCELS [26]	0.323	0.417	0.199	0.218	0.173	0.213	0.139	0.156
CMAN	**0.384**	**0.461**	**0.238**	**0.261**	**0.197**	**0.233**	**0.154**	**0.165**
CMAN-ns	0.376	0.448	0.234	0.256	0.189	0.222	0.148	0.158
CMAN-nii	0.380	0.457	0.237	0.260	0.195	0.232	0.152	0.164

In this experiment, the validity of proposed model is measured with different top-N values in Table 5, and the overall trend is similar to the previous analysis. For example, slan-ns implements 0.2608HR@5 and 0.1928NDCG@5 in Yelp, while slan-nii implements 0.2609HR@5 and 0.1940NDCG@5. Both variants of SRAN outperformed all baselines in Yelp, and RAN-NII was even more competitive than RAN-NS. The same experimental results were reflected in the Flickr dataset, confirming that both the

social network and the project-project graph contributed positively to the performance of our CMAN. Therefore, we can conclude that CMAN can improve recommendation performance by capturing high order heterogeneous information between user-user, item-item, and user-item in aggregation operations through two attention mechanisms.

4.3. Effectiveness of Our Attention Mechanisms (RQ2) and Diffusion Depth K (RQ3)

In this paper, two attention mechanisms are proposed, namely :(1) node attention block in the process of influencing diffusion; (2) Graphic attention block in the process of information aggregation. To investigate the role of these two different attention mechanisms, we compared the CMAN model with a number of model variables. We use AVG to denote an attention mechanism that degrades to equal rights of attention without any learning process. We have done some ablation studies and the results of different attention modeling combinations are shown in Table VI. In particular, we ran each submodule of SRAN with/without the corresponding attention mechanism (i.e., ATT or AVG) and found that the best performance was achieved by combining node-level attention and graph-level attention. The experimental results show that both nodes and graph attention blocks can improve the performance of the model by distinguishing important weights.

Table 6. HR@10 and NDCG@10 performance with different attentional variants (D = 64, K=2)

Graph mode	Node mode	Yelp				Flickr			
		HR	Improve	NDCG	Improve (%)	HR	Improve (%)	NDCG	Improve (%)
AVG	AVG	0.374	-	0.232	-	0.181	-	0.141	-
AVG	ATT	0.374	+0.16%	0.233	+0.56	0.181	+0.06	0.141	+0.28
ATT	AVG	0.381	+1.9%	0.237	+2.07	0.195	+7.80	0.151	+7.53
ATT	ATT	0.384	+2.7%	0.238	+2.63	0.197	+8.96	0.154	+9.38

Table 7. HR@10 and NDCG@10 performance with different diffusion depth K (D= 64)

Depth K	Yelp				Flickr			
	HR	Improve	NDCG	Improve	HR	Improve	NDCG	Improve
K = 0	0.263	- 31.40%	0.155	- 34.76%	0.079	- 59.64%	0.063	- 40.81%
K = 1	0.375	- 2.32%	0.233	- 2.14%	0.181	- 8.22%	0.142	- 7.86%
K = 2	0.384	-	0.238	-	0.197	-	0.154	-
K = 3	0.388	+1.20%	0.241	+1.13%	0.203	+3.10%	0.157	+2.01%
K=4	0.392	+2.11%	0.244	+2.48%	0.207	+5.23%	0.162	+5.39%

Next, we analyze the sensitivity of our model to the diffusion depth K and which depth value yields the best recommended results. In Table 7, we report the experimental results of SRAN in two data sets with different K values. It is worth noting that many related studies have achieved the best performance at K=2 [30,33], and performance declines as the depth of the graph continues to increase. The "Improvement" column

shows the performance change compared to the SRAN setting, that is, K=2. We found that the performance improved rapidly as K increased from 0 to 1, while the performance still improved slightly as the diffusion depth continued to increase. We conclude that the application of these two newly proposed attention mechanisms alleviates the over-smoothing problem in graph neural network training and preserves the differences in the representation of the hidden layer of each node.

5. Conclusions

In this paper, a novel framework CMAN is proposed, which effectively recommends relevant items to users. Compared with the existing algorithm, this paper first to trust matrix decomposition, avoid Pearson correlation or cosine distance must have a common object, at the same time, fully consider the trust relationship implicit similarity, improve the accuracy of sparse data, finally, this paper jointly consider the score similarity and trust relationship implicit similarity on user similarity, further improve the recommendation accuracy. This paper focuses on the improvement of trust relationship implicit similarity to social recommendation algorithms. Other related issues, such as directed trust delivery, time drift, and product characteristics, will be further explored in future work.

Acknowledgment. This work is supported by the 111 Project (B12018); Innovative Research Foundation of Ship General Performance (14422102).

References

1. Chen, X., et al.: DNNOff: Offloading DNN-based Intelligent IoT Applications in Mobile Edge Computing. IEEE Transactions on Industrial Informatics, 612-623. (2021)
2. Wang, X., et al.: Neural graph collaborative filtering. in Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. 1208-1313. (2019)
3. Zhang, J., et al. STAR-GCN: Stacked and Reconstructed Graph Convolutional Networks for Recommender Systems. in IJCAI, 612-623. (2019)
4. Chen, X., Li, M., Hsu, C. H.: A Reinforcement Learning Empowered Feedback Control System for Industrial Internet of Things. IEEE Transactions on Industrial Informatics, 1109-1124. (2021)
5. Lin, B., Huang, Y., Zhang, J. S., Hu, J., Li, J.: Cost-Driven Offloading for DNN-based Applications over Cloud, Edge and End Devices. IEEE Transactions on Industrial Informatics, Vol. 16, No. 8, 5456-5466. (2020)
6. Lin, B., Zhu, F., Zhang, J. S., Chen, J., Mauri, J. L.: A Time-driven Data Placement Strategy for a Scientific Workflow Combining Edge Computing and Cloud Computing. IEEE Transactions on Industrial Informatics, Vol. 15, No. 7, 4254-4265. (2019)
7. Do, M.P.T., Nguyen, D. V. and Nguyen, L.: Model-based approach for Collaborative Filtering. in Model-based Approach for Collaborative Filtering. 390-407. (2010)
8. Jenatton, R., et al.: A latent factor model for highly multi-relational data. in International Conference on Neural Information Processing Systems. 249-275.(2012)

9. Chen, X., et al.: Resource Allocation for Cloud-based Software Services Using Prediction-Enabled Feedback Control with Reinforcement Learning. IEEE Transactions on Cloud Computing, 111-125. (2022)
10. Rendle, S.: Factorization Machines. in ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17, December (2010)
11. Rendle, S., et al.: BPR: Bayesian personalized ranking from implicit feedback. in UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June, 18-21. (2009)
12. Rendle, S.: Factorization Machines with libFM. Acm Transactions on Intelligent Systems & Technology, 34-46, December (2012)
13. Zhou, X., et al.: SVD-based incremental approaches for recommender systems. Journal of Computer and System Sciences. Vol. 81, No. 4, 717-733. (2015)
14. Koren, Y., Bell, R. and Volinsky, C.: Matrix factorization techniques for recommender systems. Computer. Vol. 42, No. 8, 30-37. (2009)
15. Sarwar, B., et al.: Application of dimensionality reduction in recommender system-a case study. 2000, Minnesota Univ Minneapolis Dept of Computer Science,Montreal (2000)
16. Fan, M.: Non-negative matrix factorization and clustering methods application research in personalized recommendation system. East China Jiaotong University (2012).
17. Wang, X., et al.: Social recommendation with strong and weak ties. in Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, 25-37. (2016)
18. Tang, J., Hu, X. and Liu, H.: Social recommendation: a review. Social Network Analysis and Mining. Vol. 34, No. 4, 1113-1133. (2013)
19. Ma, H., et al.: social recommendation using probabilistic matrix factorization. in Proceedings of the 17th ACM conference on Information and knowledge management. 41-49. (2008).
20. Yang, B., et al.: Social collaborative filtering by trust. IEEE transactions on pattern analysis and machine intelligence. Vol. 39, No. 8, 1633-1647. (2016)
21. Fang, H., Bao, Y., and Zhang, J.: Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation. in Proceedings of the AAAI Conference on Artificial Intelligence. 11-18. (2014)
22. Tang, J., et al.: Exploiting local and global social context for recommendation. in IJCAI. 2013. Citeseer (2013)
23. Koren, Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. in Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 61-73. (2008)
24. Guo, G., Zhang, J., Yorke-Smith, T., Trustsvd, N.: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. in Proceedings of the AAAI Conference on Artificial Intelligence. 458-465. (2015)
25. Zhou, J., et al.: Graph neural networks: A review of methods and applications. AI Open. Vol. 1, 57-81. (2020)
26. Chen, S., et al.: Learning Contrastive Embedding in Low-Dimensional Space. Annual Conference on Neural Information Processing Systems. 324-332. (2022)
27. Velickovic, P., et al.: GRAPH ATTENTION NETWORKS. stat. Vol. 10, No. 50, 41-57. (2018)
28. Hamilton, W.L., Ying, R., and Leskovec, J.: Inductive representation learning on large graphs. in Proceedings of the 31st International Conference on Neural Information Processing Systems. 1231-1238. (2017)
29. Abualigah, L., Diabat, A., Mirjalili, S., et al.: The Arithmetic Optimization Algorithm. Computer Methods in Applied Mechanics and Engineering, 376-389. (2021)
30. Ying, R., et al.: Graph convolutional neural networks for web-scale recommender systems. in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 136-145. (2018)

31. Fan, W., et al.: Graph neural networks for social recommendation. in The World Wide Web Conference. 458-468. (2019)
32. Wu, L., et al.: A neural influence diffusion model for social recommendation. in Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval. 1109-1129. (2019)
33. Wu, L., et al.: Diffnet++: A neural influence and interest diffusion network for social recommendation. IEEE Transactions on Knowledge and Data Engineering, 36-45. (2020)
34. Sarwar, B., et al.: Item-based Collaborative Filtering Recommendation Algorithms. ACM, 121-137. (2001)
35. Wu, Q., et al.: Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. in The World Wide Web Conference. 21-27. (2019)
36. Tong, Z., Mcauley, J. and King, I.: Leveraging Social Connections to Improve Personalized Ranking for Collaborative Filtering. in the 23rd ACM International Conference. 1-8. (2014)
37. Chen, J., et al.: Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention. in International Acm Sigir Conference. 511-523. (2017)
38. You, D., et al.: Attributed Multi-Relational Attention Network for Fact-checking URL Recommendation. in The 28th ACM International Conference on Information and Knowledge Management (CIKM). 1358-1371. (2019)
39. He, X., et al.: Neural collaborative filtering. in Proceedings of the 26th international conference on world wide web. 590-602. (2017)
40. Rendle, S., et al.: Bayesian personalized ranking from implicit feedback. in Proc. of Uncertainty in Artificial Intelligence. 1246-1258. (2014)
41. Jamali, M., Ester, M.: A matrix factorization technique with trust propagation for recommendation in social networks, in Proceedings of the fourth ACM conference on Recommender systems. 2010, Association for Computing Machinery: Barcelona, Spain. 135–142. (2010)
42. Jiang, M., et al.: Scalable Recommendation with Social Contextual Information. IEEE Transactions on Knowledge & Data Engineering. Vol. 26, No. 11, 2789-2802. (2014)

Bin Wu received the B.Sc. degree from Jiangnan University, Wuxi, China, in 1996, and the M.S. degree from Jiangnan University, Wuxi, China, in 2005. He is currently a lecturer with the School of Internet of Things Engineering, Jiangnan University. His major research interests include visual surveillance, object detection, integrated circuit design and application of embedded system.

Tao Zhang received the bachelor's degree from Henan Polytechnic University, Jiaozuo, China, in 2008, and the Ph.D degree from the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China, in 2016. He is currently an Associate Professor with the Jiangsu Provincial Engineering Laboratory for Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi, China. He has led many research projects (e.g., the National Science Foundation and the National Joint Fund), He has authored over thirty quality journal articles and conference papers. His current research interests include data mining, information systems, wireless network, artificial intelligence, IoT and security, medical data analysis, visual surveillance, scene understanding, behavior analysis, object detection, and pattern analysis.visual monitoring, scene understanding, behavior analysis, target detection and pattern analysis.

Yeh-Cheng Chen is a Ph.D. at the Department of Computer Science, University of California, Davis, CA, USA. His research interests are radio frequency identification (RFID), data mining, social network, information systems, wireless network artificial intelligence, IoT and security.

Received: July 12, 2022; Accepted: February 10, 2023.

Computer Science and Information Systems 20(2):631–656 https://doi.org/10.2298/CSIS220717018C

Machine Learning-based Intelligent Weather Modification Forecast in Smart City Potential Area

Zengyuan Chao

Weather Modification Center of Shijiazhuang Meteorological Bureau,
Shijiazhuang, China
czy93420@163.com

Abstract. It is necessary to improve the efficiency of meteorological service monitoring in smart cities and refine the prediction of extreme weather in smart cities continuously. Firstly, this paper discusses the weather prediction model of artificial influence under Machine Learning (ML) technology and the weather prediction model under the Decision Tree (DT) algorithm. Through ML technology, meteorological observation systems and meteorological data management platforms are developed. The DT algorithm receives and displays the real meteorological signals of extreme weather. Secondly, Artificial Intelligence (AI) technology stores and manages the data generated in the meteorological detection system. Finally, the lightning monitoring system is used to monitor the meteorological conditions of Shaanxi Province from September to December 2021. In addition, the different meteorological intelligent forecast performance of the intelligent forecast meteorological model is verified and analyzed through the national meteorological forecast results from 2018 to 2019. The results suggest that the ML algorithm can couple bad weather variation with the existing mesoscale regional prediction methods to improve the weather forecast accuracy; the AI system can analyze the laws of cloud layer variation along with the existing data and enhance the operational efficiency of urban weather modification. By comparison, the proposed model outperforms the traditional one by 35.26%, and the maximum, minimum, and average prediction errors are 5.95%, 0.59%, and 3.76%, respectively. This exploration has a specific practical value for improving smart city weather modification operation efficiency.

Keywords: Artificial intelligence; Machine learning; Weather modification operation; Intelligent forecast; Decision tree

1. Introduction

With a vast territory and complicated terrain, China still lacks effective meteorological monitoring systems and technologies. For example, some meteorological stations in China fail to receive radar data occasionally due to their geographical locations, thus interrupting normal weather forecasts. Compared with the Weather Surveillance Radar (WSR), the weather events positioning system is widely used because of its extensive coverage, low maintenance cost, and long-time continuous operation [1]. In particular, atmospheric convection is often responsible for lightning, precipitation, hail, gale, and tornado, and lightning is usually followed by precipitation. Atmospheric convection can be located by referring to the location of lightning. It has been argued that precipitation

and lightning intensity have a specific correlation. Such correlation can be used to forecast the intensity of the weather conditions, which is of great significance to weather modification [2]. The lightning location data can be used to monitor convective weather and give early warning. Relevant research has shown that the echo intensity of radar well corresponds to ground flash frequencies. Positive and negative lightning frequency shows different characteristics with the storm's evolution. Generally, negative ground flash is closely related to the active period of convective weather development, and its spatial distribution corresponds to a vital updraft region and wind convergence area. In contrast, positive ground flash occurs in intense convective weather's initial and dissipation stages [3]. Lightning location is 30-60min ahead of radar echo, according to which the origin and intensity of the weather events can be calculated, and manual rain precipitation and hail prevention operations can be commanded [4]. Yet, the current weather events positioning system has shown some deficiencies. The occurrence, development, and decline of the weather events are judged by the data characteristics of lightning, providing the basis for the time selection of a weather modification. An important mechanism to improve weather modification efficiency is the real-time and accurate understanding of the distribution and transportation of precipitable water vapor in the atmosphere using lightning detection data [5]. The distribution and lightning of water vapor in weather events can be judged by its relation to lightning. The method of seeding catalysts, such as silver iodide or dry ice, is mostly used in weather modification in the convective weather process. Hence, properly-timed cloud seeding is much more cost-effective [6].

Lightning monitoring systems have been established in some areas without unified guidance on station site, equipment, communication, application, and forecast product release [7], and a full meteorological monitoring business system based on the existing system and resources will be constructed. Regarding the research of Artificial Intelligence (AI) technology in urban weather forecasting, Chang et al. (2020) used AI technology to obtain satellite image data. They verified the prediction of urban rain and flood by AI and hydraulics through 7·12 Beijing rainstorm data information [8]. Yang et al. (2019) proposed a short-term flood forecasting model based on AI technology to improve the parameter calibration, professional factors, and other problems of traditional flood forecasting models. They also processed the rainfall data of Xixian County from 2010 to 2018. It was found that the prediction duration of the model was 24 hours to 36 hours, with high prediction accuracy, meeting the requirements of flood forecasting [9].The main tasks of the lightning detection business include: (1) Strengthening the development and application research of lightning positioning technology; (2) Improving the lightning positioning system equipment; (3) Enhancing the comprehensive positioning technology of the national monitoring station network for meteorological business. Meanwhile, the performance of the meteorological monitoring station network should be evaluated, and the positioning accuracy and detection efficiency of the meteorological observation system should be improved. Thereupon, the foundation for the meteorological data application platform and shared resource databases are provided, applying the meteorological data widely while maximizing the benefits of the meteorological observation system.

The innovation of this paper is to use AI technology to establish a Local Area Network (LAN), which is used to transmit data and connect lightning detection stations. It can provide information connection for meteorological conditions in different regions. A Geographic Information System (GIS) collects and visualizes spatial data.

Multidimensional spatial data can be queried to realize data sampling. Besides, an intelligent weather forecast model for extreme weather is established through AI technology, Machine Learning (ML) technology, and the Decision Tree (DT) algorithm. The correlation between the meteorological forecast model and the monitoring and analysis system is proposed. Then, the weather forecast for extreme weather is accurate.

This exploration encompasses six sections. The first section is the introduction, explaining the importance and significance of researching intelligent meteorological forecasts, describing the problems in the current meteorological monitoring, and clarifying the research framework and main research contributions. The second section is the literature review, summarizing and analyzing the research of AI, ML, and Deep Learning (DL) methods in intelligent weather forecasting. It also defines the shortcomings of existing research and proposes the latest research algorithm. The third section describes the existing weather forecast models based on AI and ML algorithms, and the simulation parameters and environment are explained. The fourth section is the simulation analysis of the proposed model, which is compared with the other latest literature methods to verify the effectiveness of the proposed model. The fifth section is the discussion, which comparatively analyzes the results of the proposed model and previous research results. The sixth section is the conclusion, a summary pointing out the shortcomings and prospects.

2. Literature Review

2.1 AI Weather Forecast

AI refers to the simulation of human intelligence processes by machines, especially computer systems. There is a natural coupling relationship between the weather forecast, which needs massive amounts of miscellaneous data, and AI [10], which can deal with big data efficiently and infer from incomplete and uncertain information with insufficient spatial-temporal data density [11]. Additionally, AI can summarize expert knowledge and experience, improve the prediction level, and utilize an unusable understanding of statistical and numerical models. Mohammadi et al. (2018) improved the performance and efficiency of CMOS Ring Oscillator (RO) by multi-objective optimization and Particle Swarm Optimization [12]; Mohammadi et al. (2019) introduced an AIO method for modeling infinite impulse response system and evaluated the design and optimization of IIR digital filter. The results showed that the algorithm model outperformed other heuristic algorithms and Genetic Algorithms (GA) [13]; Shahraki et al. (2020) optimized the component values of analog active power filters based on multi-objective optimization. MOIPO outperformed other methods in minimizing quality factor deviation and cutting off frequency deviation [14]; Mohammadi et al. (2019) proposed an SI algorithm by introducing a novel index in IIR filter design, which was more suitable and reliable than the EC algorithm. Then, the performance of the proposed SI algorithm was analyzed from reliability, Mean Square Error (MSE), and IoS, which were better than other algorithms; Farzaneh et al. (2021)

believed that the AI method could help people plan urban construction, and AI-based modeling was widely used to predict building energy consumption [15].

2.2 ML Weather Forecast

ML, the core of AI research, primarily aims to obtain knowledge and information from input and historical data to solve more complex problems, reduce errors, and automatically learn and improve itself [16]. ML is also an interdisciplinary subject, studying how computers simulate human learning behavior to acquire and retain new knowledge and skills and reorganize the existing knowledge structure for self-improvement [17]. Kirkwood et al. (2021) believed that under ML technologies, AI could be applied to many fields, such as data mining, computer vision, natural language processing, biometrics, search engine, medical diagnosis, detection of credit card fraud, securities market analysis, DNA sequencing, speech and handwriting recognition, strategy games, and robots [18]. Hossain et al. (2020) suggested that meteorological information could be classified by DT learning in supervised learning according to the feature standard [19]. Xu et al. (2020) considered that the DT could be used to classify objects conforming to model input, which included decision points, state nodes, and result nodes. Precisely, each internal tree node (non-leaf node) and edge corresponded to an object attribute and its option. While each leaf node corresponded to the primary classification of objects [20]. Caron et al. (2020) believed that unsupervised learning could excavate the data structure and cluster input data according to feature similarities. Accordingly, the meteorological data under similar weather conditions in historical records were classified into clusters, such as gale and light rain. The current meteorological data are used for regression analysis to predict future weather events [21]. Bao et al. (2021) proposed a short-term power load forecasting model IGA-LS-SVM based on an AI algorithm. They analyzed weather, temperature, short-term power load, work, and holidays. The proposed model showed excellent predictability when associated with the existing Backpropagation algorithm, with a statistical significance of 0.8274, higher than other algorithms. Thus, the proposed IGA-LS-SVM was suitable for short-term power load forecasting [22]. Memiş et al. (2021) constructed a new classification algorithm using 18 learning real-world databases in ML, finding that FPFS-EC outperformed SVM, Fuzzy KNN, FSSC, FussCyier, HDFSSC, and BM-Fuzzy KNN in 13 of the 18 experimental datasets [23].

2.3 DL Weather Forecast

DL is a new field of ML and originates from Neural Networks (NNs) research. It can combine low-level features into more abstract high-level attribute classes or feature representations [24]. Traditionally, computer algorithms produce the desired results based on rules and data, while DL can generate appropriate rules and structures by simulating the complex human brain structure using massive amounts of training, which is especially suitable for problems that are difficult for traditional computers yet simple for human beings, such as Natural Language Understanding, handwriting recognition, and visual classification [25], as well as another extremely complex problem, weather

forecast. The weather forecast system has inevitable delays, given extensive data acquisition, conversion, and analysis. DL can be combined with the NN to analyze images directly, which has the advantages of multiple factors combination and is convenient and efficient [26].

Yin et al. (2021) considered that the WSR detected weather information images through microwave signals and displayed the echoed images on the screen, on which blue to purple indicated the echo intensity from small to large (10~70dBz). Rainfall intensity, range, possibility, and movement could be judged from different colors, color block sizes, and changes [27]. Huang et al. (2021) argued that each NN layer could calculate the color depth, color area coverage, and regional change of the image, respectively, and predict the possibility, intensity, coverage, and duration of meteorological conditions [28]. Xia et al. (2020) proposed that a satellite cloud image was an image of cloud cover and ground surface features observed from top to bottom by meteorological satellites. The Convolutional Neural Network (CNN) could identify cloud species, genera, and precipitation conditions through satellite images' color, shape, structure, brightness, and texture. For example, the deeper the green in the infrared satellite cloud image was, the stronger the ground radiation was, and the better the weather was; an arc-shaped cloud line in the image indicated an actual arc-shaped cloud line [29]. Guo et al. (2020) suggested that the CNN could determine the location and intensity of weather events according to the meteorological characteristics of these satellite images and judge their future movement and evolution trend. Image recognition could predict natural disasters, such as tsunamis, hurricanes, and thunderstorms, through radar and satellite cloud images [30]. For instance, a typhoon is caused by high ground temperature and rising hot air flow that form a low-pressure center. With the change of air pressure and the earth's movement, the air flowing into the center of low pressure also rotates, forming a counter-clockwise air vortex. The cyclone will strengthen to form a typhoon if the temperature does not drop. At present, typhoon interference cloud clusters can be identified a few days earlier, so the moving direction of a typhoon can be judged based on cloud type, wind intensity, and humidity, and its path can be released and forecasted. The advantages of the above ML methods are outstanding, but they need to be operated in the existing set-up program, while reinforcement learning can automatically adjust the state to meet the operation requirements. Table 1 is the summary and analysis of the existing literature.

Based on the above research, most scholars have conducted weather forecasting models under the development of ML and DL. However, the training of weather forecast models is greatly limited by computer hardware. Currently, there is a large amount of data for training samples of the meteorological weather forecast. Regarding time scale, the influence of weather elements at different times in recent days on future weather should be considered. On the spatial scale, there are more than ten elements in the vertical direction. Each element has more than ten levels. The accuracy of longitude and latitude grid points that can be selected within the region is also high. The data to be trained far exceeds the capacity limit of the current Central Processing Unit (CPU) after merging. The training can only be conducted after the original data is cut and magnified. The massive data resources cannot be fully utilized. ML technology is used to reform the weather forecast system to compensate for this research's lack. A meteorological monitoring business system is built to integrate meteorological monitoring, early warning and forecasting, research and technology development, and lightning detection and protection technology services. In addition, this paper uses the DT algorithm to

integrate the received thunderstorm flow meteorological signals to effectively simulate the formation of weather, which can predict extreme weather.

Table 1. The summary and analysis of the existing literature

Author (year)	Methods	Advantages	Disadvantages	Data	accuracy	Application
Mohammadi et al. (2018)	Multi-objective optimization	Better performance	Large dataset requirements	Target number	81.26%	RO
Mohammadi et al. (2019)	AIO	Data acquisition is relatively easy	It is difficult to build the model.	Pulse voltage	83.15%	Digital filter
Mohammadi et al. (2019)	SI algorithm	High robustness	Lack of stability	Pulse voltage	86.35%	IIR filter
Shahraki et al. (2020)	Multi-objective optimization	High robustness	Low accuracy	Frequency deviation	76.39%	Active power filter
Hossain et al. (2020)	DT	High accuracy	It is difficult to build the model.	Meteorological data	82.14%	Meteorological field
Xu et al. (2020)	DT + SVM	Data acquisition is relatively easy	Lack of stability	Meteorological data	80.13%	Meteorological field
Caron et al. (2020)	Unsupervised learning + clustering	High accuracy	Poor robustness	Meteorological data	83.15%	Meteorological field
Guo et al. (2020)	ML	High robustness	Low accuracy	Meteorological data	81.12%	Meteorological field
Farzaneh et al. (2021)	AI	High robustness	Low accuracy	No	76.38%	Building energy consumption
Kirkwood et al. (2021)	ML	Data acquisition is relatively easy	Lack of stability	No	No	Multiple areas
Bao et al. (2021)	AI algorithm	Better performance	Large dataset requirements	Power data	86.38%	Power load forecasting
Memiş et al. (2021)	Classification algorithm	The algorithm is easy to build.	Large dataset requirements	Multiple datasets	83.69%	Multiple areas
Yin et al. (2021)	Clustering algorithm	High accuracy	It is difficult to build the model.	Meteorological data	86.31%	Meteorological field
Huang et al. (2021)	ML	High accuracy	Lack of stability	Meteorological data	84.36%	Meteorological field

3. Intelligent Forecast Model of Weather Modification Operation

3.1 ML Weather Modification Forecast

The weather modification forecast system based on an ML algorithm includes meteorological monitoring, data storage, meteorological early warning, meteorological data sharing, weather modification enhancement, and hail suppression guidance product release. The key to a meteorological observation system is to strengthen the detection and construction of urban meteorological operations and promote the combination of district meteorological operations and weather modification scientific research and services. Finally, a meteorological monitoring business system is constructed, integrating meteorological monitoring, early warning and forecasting, research and technology development, and lightning detection and protection technology services [31]. Here, the construction demand of a regional meteorological monitoring system is analyzed explicitly from the aspects of early warning and forecasting comprehensive service based on the existing meteorological detection means and the accumulation of historical meteorological data. All kinds of data are comprehensively processed by developing a meteorological observation system and meteorological data management platform. Then, the technical means of weather modification, rain enhancement, and hail suppression are broadened based on meteorological monitoring data. Radar and satellite observation data and numerical model research are combined, and various applicable products have been developed for meteorological business and social services, thus providing visual analysis and auxiliary decision-making services for meteorological early warning and forecast and establishing a perfect region-wide weather modification forecast and early-warning business system.

The near forecast of weather modification and precipitation enhancement refers to the forecast of precipitation enhancement potential within zero to six hours. The comparison of real-time observation data is the primary decision-making basis for the forecast. The meteorological monitoring data network is used to obtain the original data of meteorological observation data in real-time and decompose the radar image combined with the observation data of radar data and satellite cloud image. The spatial-temporal changes in the weather are tracked based on the meteorological positioning, the development status is described through the exponential trend, and the early meteorological warning is realized. Figure 1 shows the structure of the meteorological early warning system. The meteorological information and the cloud images are analyzed comprehensively and used to understand the development of current weather events to facilitate weather modification. The best time for modification operation is selected according to the frequency of weather events in the cloud.

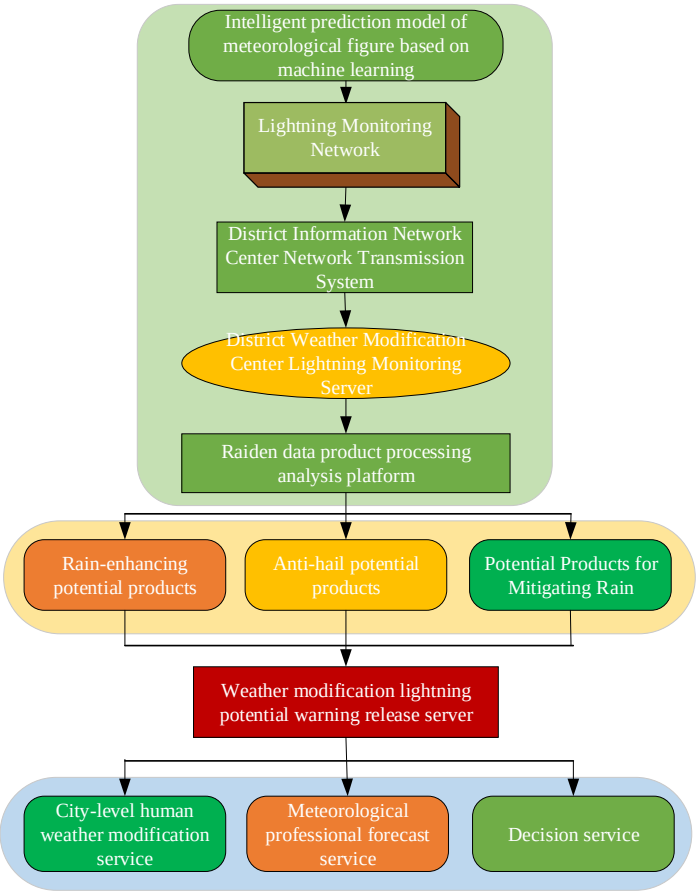


Fig. 1. Structure of meteorological early warning system

3.2 DT Weather Modification Detection

A DT is mainly used in meteorological detection algorithms. The algorithm can effectively simulate meteorological formation, generate the probability of location reliability in meteorological observation, and analyze the error of single and multi-meteorological station locations. The general process of the DT algorithm includes six aspects: collecting data, preparing data, analyzing data, constructing algorithms, testing algorithms, and applying algorithms. Any method can be used to obtain a dataset to collect data. Data often contains many errors and defects. In data preparation, feature selection is the most crucial step in data preprocessing. Attribute selection integrates subsets of data and eliminates worthless attributes. The strong correlation between attribute sets can simplify the model, reduce training time, and decrease the risk of overfitting. Analyzing the data requires selecting the appropriate statistical method to

predict the results. If not, the database needs to be adjusted. Constructing the algorithm requires building the data structure of the tree and using the training data to establish the DT model. The testing algorithm evaluates the model performance using a test data set of the constructed DT model test sample. Finally, the DT can understand the internal relationship between data and has high robustness. This means that the model can be applied to new data for data prediction and analysis. The research and application of the DT algorithm suggest that there is a one-to-one correspondence between the received meteorological signal from the thunderstorm discharge and the algorithm calculation. The model output statistical method and linear stepwise regression method are used to analyze the statistical relationship between the lightning data and the nested grid model through the frequency of weather events and the main influence of meteorological parameters. The results can be employed to estimate the possibility of regional meteorological formation. Moreover, it is pointed out that extreme weather events are often formed from large-scale stratification instability and convergence provided by humidity and local wind field. The algorithm analyzes 274 meteorological signals received by four meteorological stations and then filters and compares them step by step. The crucial factors are selected from the four stations, and a Multiple Regression (MR) equation is established to calculate the real meteorological signal for receiving and displaying. The MR equation should use a variance ratio to test the significance of the regression equation. If the influence factor passes the significance test, it can enter the equation. Otherwise, it should not enter the regression equation. The criterion for excluding variables in the regression equation is also to use the variance ratio for the significance test. The test eliminates the variables that contribute the least to the sum of squares of partial regression, whether they are selected into or excluded from the regression equation. The selection and exclusion items that meet the conditions are eliminated. The stepwise regression method eliminates the factors that have little influence on the dependent variables, reducing the difficulty of analyzing problems. It also improves the calculation efficiency and the stability of the regression equation, with good prediction accuracy. The error is minimized through the accurate calculation of the algorithm. Meanwhile, an accurate forecast is conducted, and the research on the numerical model of the weather forecast is strengthened [32].

The DT method is one of the fast and most intuitive inductive learning methods. It has been widely used in expert systems, industrial control processes, financial insurance prediction, and medical diagnosis. It can integrate various data and forecast methods in the weather potential forecast on a platform. The DT method is applied in the meteorological monitoring system to compare the data of positioning instruments, conventional radiosonde, satellite cloud image, and T213 numerical forecast products and presents real and reliable meteorological formation data [33]. Specifically, the DT algorithm should be used in the meteorological monitoring system. First, in the DT generation phase, the algorithm judges the branches of the DT according to the root of the attribute parameter value of each node. It selects the path channel attribute to distinguish the sample parameters. The distance between the branch node tree and the root node is selected. The probability of a reliable position in generating meteorological observation can be determined according to the distance between nodes. Also, the error of single and multiple meteorological station positions can be analyzed. During this period, the branch judgment standard of the DT is selected according to the lightning detection algorithm as the tradeoff standard for using the data branch information value.

The DT algorithm can better integrate all data and forecast methods in the potential meteorological forecast on one platform. It is used to establish the meteorological monitoring and forecast system. The satellite cloud image and the real meteorological data are integrated and used for weather forecasts to improve the system's performance. The binary tree algorithm is used here, and the branch information is used as the judgment standard of DT branch selection. The binary DT is established through the established rules of attribute branching. Figure 2 presents the collection process of a meteorological signal. The specific path is to judge whether the selected path is a multi-possibility selection or whether the lightning signals collected by each station are inconsistent. The multi-station comparison method is employed to select the best path for the possibility of lightning formation. The collected data point signals (>ten-point) are counted. All the points are digitized and statistically analyzed to select the following path. Each attribute value's weight is found, and the best path is selected.

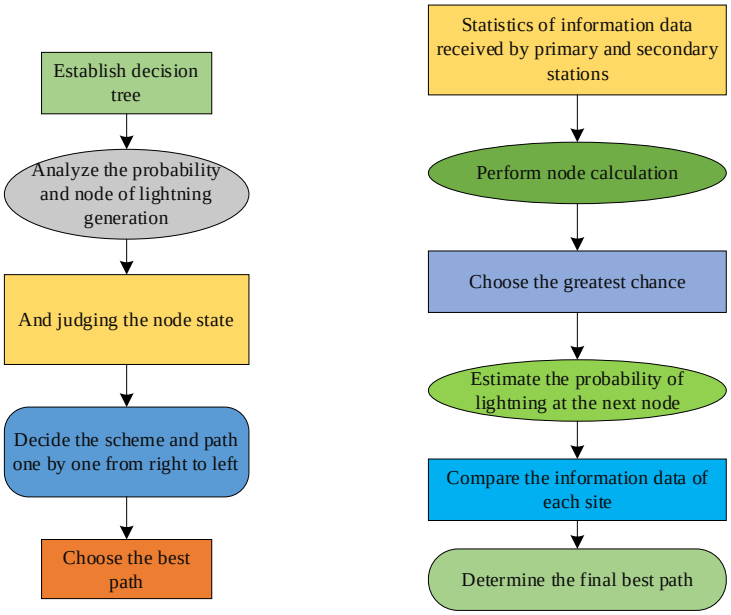


Fig. 2. Flowchart of meteorological signal collection under different conditions, (a) Flowchart of data collection when meteorological signals of each station are consistent;(b) Flowchart of data collection in case of inconsistent meteorological signals at each station

The local electronic map should be loaded into the lightning detection system. Mapinfo is used to add label layers, such as lightning locator and city names on the map, to assist in display and query. Then, the labeled map is loaded into the lightning detection system. The lightning detection system should include an essential layer for background display and an item layer for lightning positioning data. Figure 3 presents the process of loading electronic map data.

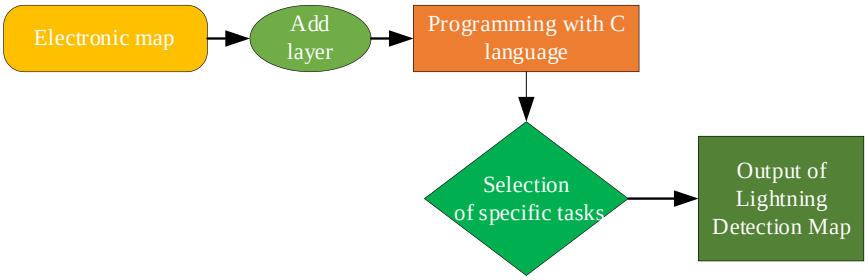


Fig. 3. Loading flow chart of electronic map data

The DT can be simplified using the impurity function φ (p1, p2, ..., p*j*) on each branch node. For any node *t*, the calculation of impurity *i*(*t*) reads:

$$i(t) = \varphi(p(1|t), p(2|t), \ldots, p(j|t)) \tag{1}$$

p(*i*|*t*) represents the probability of the node *t* samples belonging to category *i*.

The impurity reduction after branch *s* divides the parent node *t* into child nodes t_R and t_L reads:

$$\Delta i(s,t) = i(t) - P_R i(t_R) - P_L i(t_L) \tag{2}$$

P_R and P_L represent the proportion of child nodes t_R and t_L samples in parent node *t* samples, respectively. The reduction in impurity characterizes the branching effect φ (s, t).

The branching criterion: the criterion with the largest impurity reduction is used for further branching, which is expressed by Eqs. (3) and (4).

$$\Delta i(s,t) = \max_{s_j \in S} \Delta i(s,t) \tag{3}$$

$$S = \{s_1, s_2, \mathrm{K}\} \tag{4}$$

The criterion for branching termination: branching terminates when the maximum impurity reduction is less than the given threshold β.

The binary DT based on a growth-pruning method is used for error analysis of the lightning detection system. The data judgment accuracy of the original tree is not ideal because of redundant nodes. After continuous pruning and selection, the attribute of each node in the lightning detection DT is the threshold of the attribute of generating lightning weather. The contribution of different attributes greater than the threshold of the same node to the generation of lightning weather is different. The DT of lightning detection calculates the possible lightning data of each 4°×4° longitude and latitude grid point in Tibet (The geographic information map of Tibet is divided into multiple 4°×4° grid blocks). The maximum depth of the DT is five because the classification is based on the work area size and combines related references. The DT is used to analyze the possibility of lightning occurrence. The data analysis shows that the greater the value obtained is, the larger the possibility of lightning occurring in the detection point is; on

the contrary, the smaller the probability is, the less likely lightning will appear in the detection point.

3.3 AI Meteorological Data Storage

AI is a branch of computational science, which is to study how to simulate human brain thinking through the computer to realize machine intelligence. AI involves a wide range of disciplines, including both natural and social disciplines, and is mainly realized through traditional programming technology and simulation. Firstly, manual programming, as an engineering method, is tedious and error-prone and is used in character recognition and AI chess players. Secondly, the simulation method is based on biological thinking and, thus, is more complex, including GA and Artificial Neural Network algorithms. The simulation method is more robust and can handle more complex problems [34].

The introduction of AI has excellent advantages in model correction, which can significantly improve the prediction ability of temperature, precipitation, and other meteorological elements. In addition, AI technology can also track the development of weather processes. AI technology can locate typhoons according to satellite cloud maps through image recognition to tracktyphoonsn in real time, quickly, and intelligently. On the one hand, the development process of typhoons can be accurately described. On the other hand, a good initial field for the model can be provided, which helps improve the ability of the model to predict typhoons. The meteorological monitoring system will produce a large amount of data, which needs to be effectively managed and stored. Through strong learning ability and fast reasoning speed, AI technology provides a database for multi-source multi-resolution Spatio-temporal observation data and multi-mode and multi-scheme prediction data in the meteorological field. The numerical weather model forecasts and simulates the weather based on the complex atmospheric motion process. AI algorithms are driven by big data. It can capture meteorological data's temporal and spatial characteristics and nonlinear relationships. SQL Server 2000 database stores lightning location data collected through the control and data operation statements. Data collection mainly includes: reading data text format and data entry. The data entry accuracy can be intuitively understood through feedback, including the database name, database authentication method, entry time, entry name, and the entry row number. The data is initially explored, and the model is constructed. Then, they are applied to all data through loop traversal.

Data query: multiple controls and SQL query statements are used to match the query information field. The queried contents are shown in the table. The lightning data query system can draw data output to give users an intuitive understanding of the lightning data, through which the strong center of weather events can be located. The queried lightning data are used to analyze the intensity center of lightning occurrence. Then, the lightning frequency, position of the intensity center, formation direction, distance from the detection point, and the intensity of the maximum charge center are obtained. Finally, the discharge area of the thunderstorm cloud center is obtained through the corresponding equation of the meteorological precipitation enhancement potential forecast.

Here, a smart city's intelligent weather prediction model is realized through ML technology, DT algorithm, and AI technology. The role of different research methods in this paper is shown in Table 2.

Table 2. Functions of research methods in this paper

Research method	Function
AI technology	Effectively store and manage data of meteorological monitoring system.
ML	The meteorological forecast system can be reformed. A meteorological monitoring business system can be established by integrating meteorological monitoring, early warning and forecasting, research and technology development, and lightning detection and protection technology services.
DT algorithm	Collect meteorological signals under different conditions.

3.4 System Parameters and Environment Settings

The network environment in the lightning monitoring system is based on the LAN and the General Packet Radio Service wireless network. The software architecture is established through the Client/Server technology network based on Windows 10 operating system. The lightning monitoring system is designed using the principle of function separation, independent work, non-interference, and scalability. The system includes a display unit for lightning data, a receiving unit for secondary station data, a multi-station positioning unit, and a data viewing unit. The receiving unit for the secondary station data can be expanded multiple times without changing the source program. It has strong adaptability to the increase or decrease in the number of secondary stations.

Background running environment: the hardware of the application server is HP PROLIANT DL580G7, equipped with 4 Xeon E7-4807 1.86GHzCPU, 16G DDR3 memory, 900G hard disk, configured with Windows 10 ServerR2 operating system, and JBoss and GeoServer. The database server is HP PROLIANT DL580G7, equipped with 4 Xeon E7-4807 1.86GHzCPU, 8G DDR3 Memory, and 900G hard disk, configured with Windows2008ServerR2 operating system, and PostgreSQL and SQLServer2005 [35].

3.5 Experimental Data

The data here comes from the National Climatic Data Center (NCDC). The lightning meteorological conditions from September to December 2021 in Shaanxi Province and the meteorological forecast results of Yanta District in Xi'an, Shaanxi Province, from 2018 to 2019 are retrieved through data. Table 3 reveals the data components of meteorological variables.

Table 3. Data composition of meteorological variables (data source: NCDC)

Name	Root Mean Square Error	Deviation	Correlation coefficient
Air temperature	0.88K	-0.13K	0.97
Specific humidityc	4.76%	4.76%	0.93
Wind speed	0.83m/S	-0.21%	0.82
Surface barometric pressure	3.74hPa	-0.38hPa	0.96
Hourly precipitation	0.94mm/H	-0.004 mm/H	0.72

Additionally, the number of input features and the length of output results of each sample must be the same to reflect that the intelligent prediction model of a smart city based on AI and ML does not need all training samples to maintain consistency. This is also the characteristic of the verification model proposed here, which is different from other ML algorithms. The number of training set data is 300, and the number of test set data is 100. Training data set samples are input in the training phase, n categories. The number of samples of each type and the number of samples of the maximum type m are recorded. The random list L of each category is used to modularize the number of category samples. Resampling is performed according to the modular operation results so that the number of samples of each class is m. Its index is stored in the corresponding L. Finally, all the L lists are combined to form a final list and output the training list. Secondly, the intelligent city prediction model based on AI and ML proposed here is developed for meteorological prediction, so its evaluation algorithm is more specific and rigorous than the general algorithm. The real-time test set of the algorithm is the real-time weather data collected every day during the test period and the results observed by multiple observers back-to-back, which cannot be used for model verification and evaluation in advance. In model training and testing, AI technology is used to store and manage the data of the meteorological monitoring system. The DT algorithm is used to collect meteorological signals under different conditions. This can effectively avoid the problem of model applicability caused by training data sets. Based on AI technology and ML, the overall consistency can be observed for a long time under different weather conditions in different regions. The model's applicability to different geographical locations is also tested while testing the consistency with manual observation.

4. Model Simulation and Comparative Analysis

4.1 Analysis of Meteorological Statistics

Here, the frequency of lightning occurrence in Yanta District, Xi'an City, Shaanxi Province, is statistically analyzed to conduct statistical analysis on the lightning monitoring system and verify its feasibility of the lightning monitoring system. The

frequency change of a lightning occurrence in Yanta District, Xi'an City, Shaanxi Province is shown in Figure 4. The results reveal that the lightning monitoring system can calculate the lightning flash frequency in a cycle and draw it out through the broken line chart. It can well reflect the changing trend of the relevant data during this period and can intuitively show the monthly lightning frequency, which is conducive to intuitive and concise analysis. The frequency of lightning in September is significantly higher than that in other months, and that in December is only 0.2.

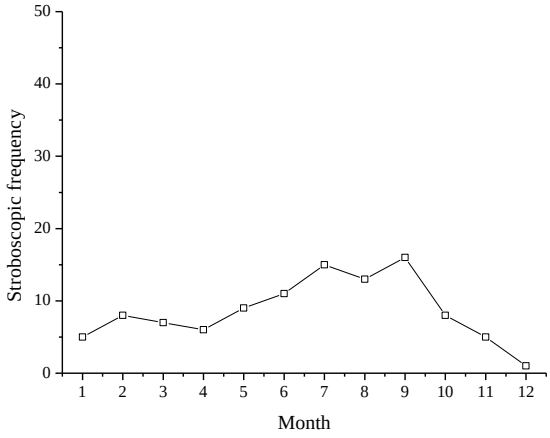


Fig. 4. Frequency change of a lightning occurrence in Yanta District, Xi'an City, Shaanxi Province

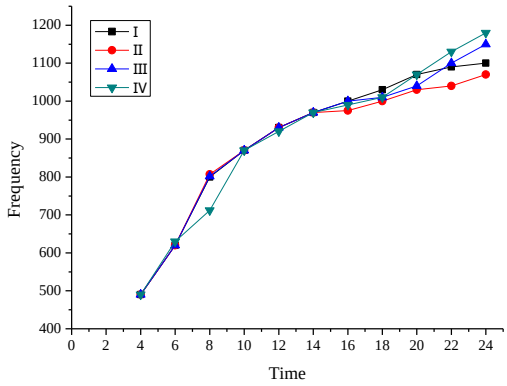


Fig. 5. Statistical chart of lightning frequency in different areas of a day\

In Figure 5, the lightning frequency in different areas (Xi'an: I, Hanzhong: II, Shangluo: III, and Weinan: IV) is calculated and plotted. It is found that the combination of lightning frequency and satellite cloud image information can be used to obtain the law of lightning occurrences in the future. The lightning frequency significantly increases with time, and the highest lightning frequency in IV is 1,163

times in 24 hours. Accordingly, the right time for weather-modification operations can be determined.

4.2 Performance analysis of intelligent forecasting

In Figure 6, two forecast dates in 2018 [36] and 2019 [37] are used as examples. The dataset of the four forecasting models is the meteorological forecast data of Yanta District, Xi'an City, Shaanxi Province, from 2018 to 2019. The different meteorological intelligent forecasting performances in the Yanta District of Xi'an City are predicted. Figure 6A shows the forecast results for 2018, revealing that under the training set and the test set, the forecast results of the model deviate greatly from the actual results. The main reason is the great fluctuation of China's meteorological situation in 2018 and the incomplete dataset, which leads to insufficient learning performance. In the meteorological intelligent forecast results in 2019 (Figure 6B), the model's performance is consistent with the actual meteorological results in different sets, but there is a short-term fluctuation at 10-20 o'clock. Thus, the weather intelligent forecast system based on the ML algorithm performs well but relies on specific data.

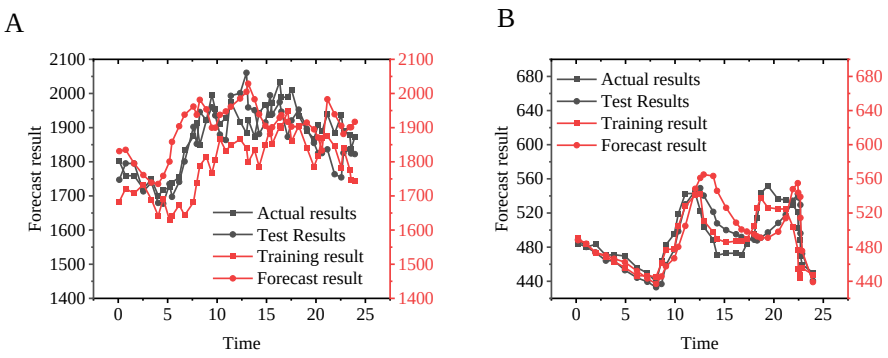


Fig. 6. Results of meteorological intelligent forecast performance

4.3 Comparative analysis of model performance

The performance of an intelligent city prediction model based on AI and ML is verified to analyze the performance results of the meteorological intelligent prediction model. The training data set is the weather forecast data of Xi'an on September 4, 2021. Figure 7 illustrates the performance results of the meteorological intelligent forecast model. The performance analysis of different methods suggests that the performance of the model is reduced by 20.35% without an AI management system, while the performance of the model without a DT algorithm is reduced by 15.63%. Without AI and DT algorithms, the model's performance is reduced by 35.68%. The performance of the proposed model is improved by 35.26% compared with the traditional model. The performance of the latest literature models [38-41] is compared with that of the

proposed model. Compared with the literature model in reference 23, the performance of the proposed model is slightly weaker, but it has a significant improvement compared with the model in other references. This further shows the effectiveness of the proposed weather intelligent forecast model, which has good forecast accuracy. The accuracy of the proposed model is not as good as that in reference 23, but the proposed model has much lower requirements for the computer configuration. Therefore, under the same equipment, the accuracy of the proposed model can exceed that in most literature. It is believed here that DL has a broader application for meteorological modeling in the future.

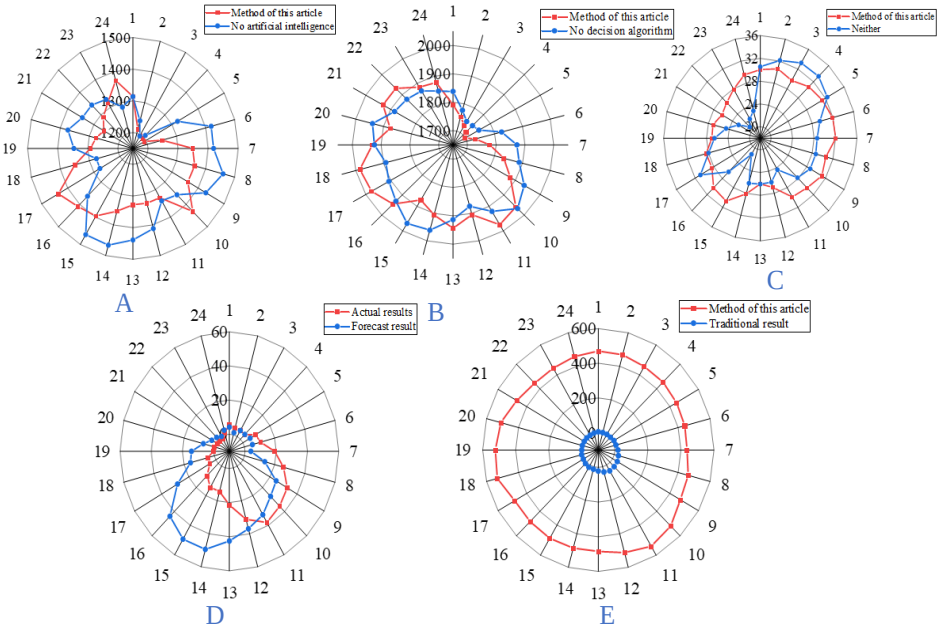


Fig. 7. Performance comparison analysis of different models((A) shows the performance comparison between the research method in this paper and the model without AI technology; (B) indicates the performance comparison between the research method in this paper and the model without DT algorithm; (C) represents the performance comparison between the research method in this paper and the model without AI technology and DT algorithm; (D) represents the comparison between the actual meteorological forecast results and the model prediction results; (E) shows the performance comparison between the research method in this paper and the traditional prediction model)

4.4 Performance Comparison of Different Models

Further, to compare the performance of the intelligent city prediction model based on AI and ML, the proposed model is compared with literature models in references 36-37 from Accuracy, Precision, Recall, and F1 score. The calculation reads:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} . \quad (5)$$

In Eq.(5), $TP + TN$ represents the number of samples predicted correctly, and $TP + TN + FP + FN$ stands for the total number of samples.

$$\Pr ecision = \frac{TP}{TP + FP} . \quad (6)$$

In Eq.(6), TP represents the number of samples correctly predicted as 1, and $TP + FP$ denotes the total number of samples predicted as 1.

$$\mathrm{Re}call = \frac{TP}{TP + FN} . \quad (7)$$

In Eq.(7), $TP + FN$ stands for the number of samples predicted to be 1 in the real case.

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} . \quad (8)$$

The F1 value is a harmonic average of model accuracy and recall.
The results are shown in Figure 8.

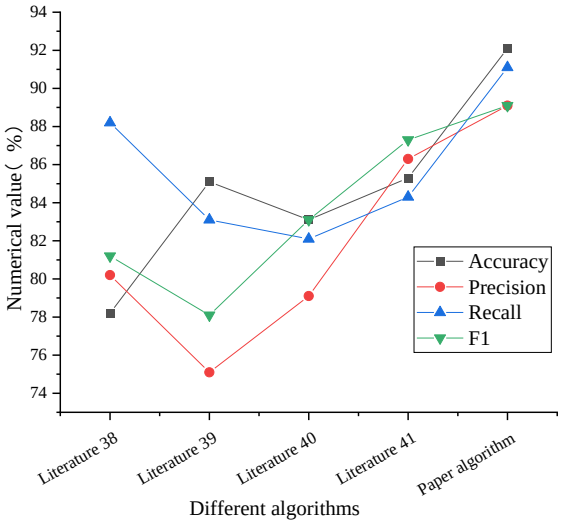


Fig. 8. Comparison of recognition accuracy of different models

Figure 8 tells that the recognition accuracy of the proposed intelligent weather forecast model has reached 92.1%, which is at least 15.1% higher than that of other models. At the same time, the Precision, Recall, and F1 score of the proposed model are

also the highest, indicating that compared with other intelligent weather forecast models, the proposed model has better performance and higher prediction accuracy.

4.5 Analysis of actual forecasting performance

The meteorological forecast data of Xi'an from September 4 to September 30, 2021 is taken as the training data set to analyze the performance of the intelligent forecasting model of smart city integrated with AI and ML. Figure 9 demonstrates the forecast analysis results of the actual performance of the proposed model. It suggests that the accuracy of meteorological intelligent forecast results within one month is basically maintained at 96%, of which the maximum, minimum, and average forecast errors are 5.95%, 0.59%, and 3.76%, respectively. It proves that the proposed model has high forecast accuracy and good robustness, which can meet the urban daily weather forecast needs.

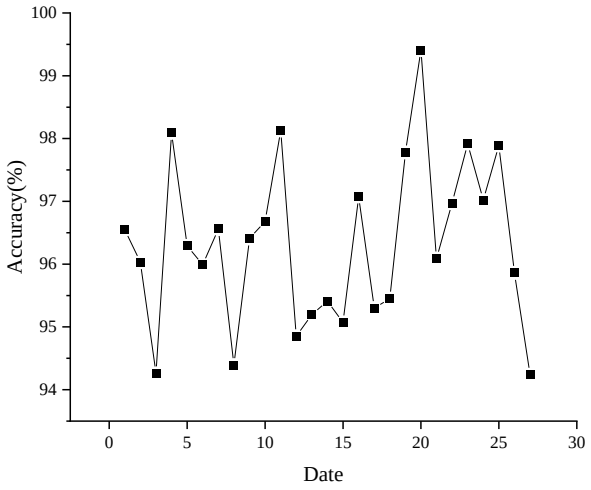


Fig. 9. Analysis results of actual forecast performance of the proposed model

Table 4. Analysis of meteorological early warning results

Thunderstorm Serial number	Lightning station Current location	Distance	Movement				Prediction accuracy
M0	143	148	0min	15min	30min	45 min	(%)
F0	132	165	2/3	44/147	NO	NO	91%
H0	145	165	NO	129/135	NO	NO	92.45%
D0	120	155	189/5	144/155	143/155	NO	91.35%
O0	95	99	235/7	118/134	115/134	NO	90.21%
P0	141	163	NO	NO	NO	NO	90.13%
MO	113	125	120/145	NO	NO	NO	92.54%

Table 4 displays the results of meteorological location early warning. The positions of seven meteorological images are analyzed, and the information of the lightning station determines the specific location information of these points. The actual data will be used to analyze the proposed model and compared with the existing data to obtain the final prediction result. From the meteorological early warning results in Table 4, the model has apparent early warning information within 0~30 minutes. After 45 minutes, the warning information disappears. Moreover, the model's average prediction accuracy in seven images is 91%, proving the proposed model's effectiveness.

5. Discussion

The meteorological observation and early warning system study focuses on some of the leading image display core technologies in the development of GIS. The system adopts Mapinfo, MapX graphics development software, and a system application platform. Meanwhile, the enterprise user database platform based on SQL Server 2000 is designed and built. Many functions are realized, including the integrated analysis and retrieval of the monitoring attributes and spatial data of the actual lightning location data, the spatial distribution statistics of the lightning location data, and the visualization grid demonstration of charts. It facilitates the early warning and forecasting personnel of weather modification, precipitation enhancement, and hail suppression to provide services for the weather forecast and public users. Meteorological monitoring, early warning, and forecast are realized. The application of the ML algorithm improves the forecast accuracy of meteorological monitoring system data and realizes the data classification while simplifying data processing and improving the system calculation efficiency. In Reference [42], scholars also applied ML algorithms to construct intelligent weather forecasts and built an intelligent weather forecast system. The algorithm could help the relevant personnel intuitively analyze the weather changes to improve the forecast accuracy. This is consistent with the idea of this exploration.

Further, the relevant forecast factors are screened using various numerical products through the DT method. Then, the DT of the potential lightning forecast is established with the monitoring data of the lightning locator network, high altitude ground observation data, wind field data, and radar data of the Tibet Autonomous Region. The potential meteorological forecast with a spatial resolution of 1°*1° and temporal resolution of 1 hour is realized. The weather events positioning system is combined with the domestic mature satellite cloud images and weather detection system data, the occurrence and change process of meteorological phenomena are analyzed, and the temporary weather forecast is realized. In Reference [43][44], scholars also established a temporary early-warning system through an ML algorithm to help modify operations and predict weather phenomena, achieving high forecast accuracy. The results are consistent with the temporary weather early warning system proposed, proving that the proposed meteorological monitoring system is feasible, especially in the areas with imperfect meteorological monitoring systems, to improve the data processing technology of local meteorological monitoring, early warning, and forecasting products. Additionally, the proposed system can improve their fineness, visibility, evaluability, and forecast accuracy and help the weather modification operation of relevant personnel, such as precipitation enhancement and hail suppression.

6. Conclusion

Given the imperfection of meteorological monitoring, early warning, and forecasting in urban weather modification centers, targeted research is conducted here. Based on the GIS, the potential forecast model of meteorological precipitation enhancement monitoring is implemented together with a temporary early-warning system. Various data structures are analyzed in connection with the spatial data, and an intuitive and visual query analysis method is established to calculate weather formation in the spatial range. The calculation and forecast results are verified by real-time visual data regarding the thunderstorm weather. Through software development, the meteorological detection system has been put into trial operation. The main achievements of this paper are as follows.

1)After repeated operation and testing, the system is stable. Its function fully meets the design requirements of urban weather modification, precipitation enhancement, and hail suppression early warning and prediction, improving early warning and prediction in the region.

2)The GIS technology is applied to the data processing technology of lightning monitoring, early warning, and forecasting products, significantly improving the fineness, visualization, evaluability, and prediction accuracy. Mainly, the research involves the embedment of observation data of the new generation Doppler WSR, the surface precipitation data, and the early warning results into the GIS platform.

3)Under comparison analysis, the monitoring and forecasting results are more in line with the actual situation.

Still, there are some deficiencies. First, there is no perfect meteorological database. The existing data content of domestic scientific research institutions is only used for internal testing and is not fully open access. Moreover, NN or DL content can be used for effective data processing for the proposed model. In the follow-up study, these two aspects will be improved, and the model performance will be optimized.

The research contributions read: (1) The collected meteorological data are used for precipitation enhancement, hail suppression potential prediction, and meteorological early warning products. (2) A metropolitan LAN is established to transmit data and connect lightning detection stations based on regression algorithm, DT algorithm, and batch processing method. (3) Two main functional modules are evaluated from two aspects: loading a GIS electronic map and superimposing radar and cloud map and high-altitude data based on AI technology. A background database based on meteorological observation is established. It can collect and visualize GIS and query multidimensional spatial data to achieve data sampling. (4) The 24-hour meteorological monitoring technology is studied through comparison and superposition. Relevant forecast factors are screened to form potential forecast data for weather modification. The meteorological potential prediction products are built on the meteorological data analysis platform to monitor meteorological conditions, such as spatial location, distribution location, and time resolution.

Acknowledgment. The author acknowledge the help from the university colleagues.

References

1. Hu H., Lei T., Hu J., Zhang S., Kavan P. Disaster-mitigating and general innovative responses to climate disasters: Evidence from modern and historical China. International Journal of Disaster Risk Reduction, Vol. 28, 664-673. (2018)
2. Li R., Jin Y. The early-warning system based on a hybrid optimization algorithm and fuzzy synthetic evaluation model. Information Sciences, Vol. 435, 296-319. (2018)
3. Zhou W., Berrio J.S., Worrall S., Nebot E. Automated evaluation of semantic segmentation robustness for autonomous driving. IEEE Transactions on Intelligent Transportation Systems, Vol. 21, No. 5, 1951-1963. (2019)
4. Song T., Zhao H., Liu Z., Liu H., Hu Y., Sun D. Intelligent human hand gesture recognition by local–global fusing quality-aware features. Future Generation Computer Systems, Vol. 115, 298-303. (2020)
5. Marshall A., Ojiako U., Wang V., Lin F., Chipulu M. Forecasting unknown-unknowns by boosting the risk radar within the risk intelligent organization. International Journal of Forecasting, Vol. 35, No. 2, 644-658. (2019)
6. Wang B., Gu X., Ma L., Yan S. Temperature error correction based on BP neural network in the meteorological wireless sensor network. International Journal of Sensor Networks, Vol. 23, No. 4, 265-278. (2017)
7. Yogavani D., Prakash N.K. Implementation of wireless sensor network-based multi-core embedded system for the smart city. International Journal of Computer Technology and Applications (IJCTA), Vol. 10, No. 02, 119-123. (2017)
8. Chang D L, Yang S H, Hsieh S L, et al. Artificial intelligence methodologies applied to prompt pluvial flood estimation and prediction. Water, Vol. 12, No. 12, 3552. (2020)
9. Yang T, Sun F, Gentine P, et al. Evaluation and machine learning improvement of global hydrological model-based flood simulations. Environmental Research Letters, Vol. 14, No. 11, 114027. (2019)
10. Schwendicke F.a., Samek W., Krois J. Artificial intelligence in dentistry: chances and challenges. Journal of dental research, Vol. 99, No. 7, 769-774. (2020)
11. Schneider P., Walters W.P., Plowright A.T., Sieroka N., Listgarten J., Goodnow R.A., et al. Rethinking drug design in the artificial intelligence era. Nature Reviews Drug Discovery, Vol. 19, No. 5, 353-364. (2020)
12. Mohammadi A., Mohammadi M., Zahiri S.H. Design of optimal CMOS ring oscillator using an intelligent optimization tool. Soft Computing, Vol. 22, No. 24, 8151-8166. (2018)
13. Mohammadi A., Zahiri S.H., Razavi S.M. Infinite impulse response systems modeling by artificial intelligent optimization methods. Evolving Systems, Vol. 10, No. 2, 221-237. (2019)
14. Shahraki N.S., Zahiri S.H., editors. Multi-objective optimization algorithms in analog active filter design. 2020 8th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS); 1258-1261. (2020)
15. Farzaneh H., Malehmirchegini L., Bejan A., Afolabi T., Mulumba A., Daka P.P. Artificial Intelligence Evolution in Smart Buildings for Energy Efficiency. Applied Sciences, Vol. 11, No. 2, 763-773. (2021)
16. Lei Y., Yang B., Jiang X., Jia F., Li N., Nandi A.K. Applications of machine learning to machine fault diagnosis: A review and roadmap. Mechanical Systems and Signal Processing, Vol. 138, 106587-106593. (2020)
17. Dral P.O. Quantum chemistry in the age of machine learning. The journal of physical chemistry letters, Vol. 11, No. 6, 2336-2347. (2020)
18. Kirkwood C., Economou T., Odbert H., Pugeault N. A framework for probabilistic weather forecast post-processing across models and lead times using machine learning. Philosophical Transactions of the Royal Society A, Vol. 379, No. 2194, 20200099-20200106. (2021)
19. Hossain M.S., Mahmood H. Short-term photovoltaic power forecasting using an LSTM neural network and synthetic weather forecast. IEEE Access, Vol. 8, 172524-172533. (2020)

20. Xu W., Ning L., Luo Y. Wind speed forecast based on post-processing of numerical weather predictions using a gradient boosting decision tree algorithm. Atmosphere, Vol. 11, No. 7, 738-742. (2020)
21. Caron M., Misra I., Mairal J., Goyal P., Bojanowski P., Joulin A. Unsupervised learning of visual features by contrasting cluster assignments. arXiv preprint arXiv:200609882, 26-31. (2020)
22. Bao D L., Xin Y Z., Mei Z., Hui L., Guang-Qian L. Improved genetic algorithm-based research on optimization of least square support vector machines: an application of load forecasting. Soft Computing: 1-9. (2021)
23. Memiş S., Enginoğlu S., Erkan U. Numerical Data Classification via Distance-Based Similarity Measures of Fuzzy Parameterized Fuzzy Soft Matrices. IEEE Access, Vol. 9, 88583-88601. (2021)
24. Tian C., Fei L., Zheng W., Xu Y., Zuo W., Lin C.W. Deep learning on image denoising: An overview. Neural Networks, 124-131. (2020)
25. Bahri Y., Kadmon J., Pennington J., Schoenholz S.S., Sohl-Dickstein J., Ganguli S. Statistical mechanics of deep learning. Annual Review of Condensed Matter Physics, 224-233. (2020)
26. Minaee S., Boykov Y.Y., Porikli F., Plaza A.J., Kehtarnavaz N., Terzopoulos D. Image segmentation using deep learning: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 115-163. (2021)
27. Yin X., Hu Z., Zheng J., Li B., Zuo Y. Study on Radar Echo-Filling in an Occlusion Area by a Deep Learning Algorithm. Remote Sensing, Vol. 13, No.9, 1779-1783. (2021)
28. Huang C.J., Shen Y., Chen Y.H., Chen H.C. A novel hybrid deep neural network model for short-term electricity price forecasting. International Journal of Energy Research, Vol. 45, No. 2, 2511-2532. (2021)
29. Xia M., Tian N., Zhang Y., Xu Y., Zhang X. Dilated multi-scale cascade forest for satellite image classification. International Journal of Remote Sensing, Vol. 41, No. 20, 7779-800. (2020)
30. Guo Y., Cao X., Liu B., Gao M. Cloud Detection for Satellite Imagery Using Attention-Based U-Net Convolutional Neural Network. Symmetry, Vol. 12, No. 6, 1056-1063. (2020)
31. Khalyasmaa A.I., Eroshenko S.A., Tashchilin V.A., Ramachandran H., Piepur Chakravarthi T., Butusov D.N. Industry Experience of Developing Day-Ahead Photovoltaic Plant Forecasting System Based on Machine Learning. Remote Sensing, Vol. 12, No. 20, 3420-3426. (2020)
32. Benkercha R., Moulahoum S. Fault detection and diagnosis based on C4. 5 decision tree algorithm for the grid-connected PV system. Solar Energy, Vol. 173, 610-634. (2018)
33. Ali E.M., Ahmed M.M., Wulff S.S. Detection of critical safety events on freeways in clear and rainy weather using SHRP2 naturalistic driving data: Parametric and non-parametric techniques. Safety Science, Vol. 119, 141-149. (2019)
34. Landeras G., Bekoe E., Ampofo J., Logah F., Diop M., Cisse M., et al. New alternatives for reference evapotranspiration estimation in West Africa using limited weather data and ancillary data supply strategies. Theoretical and Applied Climatology, Vol. 132, No. 3, 701-716. (2018)
35. Bu J., Peng D.D., Xiao H., Yue Q., Han Y., Lin Y., et al. Analysis of meteorological conditions and prediction of epidemic trend of 2019-nCoV infection in 2020. MedRxiv, 1211-1234. (2020)
36. Işık E., Inallı M. Artificial neural networks and adaptive neuro-fuzzy inference systems approaches to forecast the meteorological data for HVAC: The case of cities for Turkey. Energy, Vol. 154, 7-16. (2018)
37. Fuente A., Meruane V., Meruane C. Hydrological early warning system based on a deep learning runoff model coupled with a meteorological forecast. Water, Vol. 11, No. 9, 1808. (2019)

38. Kang H., Hong T., Lee M. A new approach for developing a hybrid sun-tracking method of the intelligent photovoltaic blinds considering the weather condition using data mining technique. Energy and Buildings, Vol. 209, 109708-109712. (2020)
39. Li C., Xu P. Application on traffic flow prediction of machine learning in intelligent transportation. Neural Computing and Applications, 1-12. (2020)
40. Kuhn P., Wirtz M., Wilbert S., Bosch J., Wang G., Ramirez L., et al. Field validation and benchmarking of a cloud shadow speed sensor. Solar Energy, Vol. 173, 229-45. (2018)
41. Narote S.P., Bhujbal P.N., Narote A.S., Dhane D.M. A review of recent advances in lane detection and departure warning system. Pattern Recognition, Vol. 73, 216-34. (2018)
42. Pallonetto F., De Rosa M., Milano F., Finn D.P. Demand response algorithms for smart-grid ready residential buildings using machine learning models. Applied Energy, Vol. 239, 1265-82. (2019)
43. Barton C., Chettipally U., Zhou Y., Jiang Z., Lynn-Palevsky A., Le S., et al. Evaluation of a machine learning algorithm for up to 48-hour advance prediction of sepsis using six vital signs. Computers in biology and medicine, Vol. 109, 79-84. (2019)
44. Shen, C.-w., Min, C., & Wang, C.-c. (2019). Analyzing the trend of O2O commerce by bilingual text mining on social media. Computers in Human Behavior, Vol. 101, 474-483. https://doi.org/10.1016/j.chb.2018.09.031

Appendix.

Table 3. Variable information

Variables	Definition	Calculation
$p(i\|t)$	It indicates the proportion of samples with node number t belonging to category i.	It is calculated by Eq. (1).
s	Number of branches	Fixed value
t_R	Number of R branches	Fixed value
t_L	Number of L branches	Fixed value
P_R	The proportion of sample tR in the sample with node t.	It is calculated by Eq. (2).
P_L	The proportion of sample tL in the sample with node t.	It is calculated by Eq. (2).
β	Threshold size	Fixed value

Table 4. English abbreviations

Abbreviations	Full name	Meaning
CMOS	Complementary Metal Oxide Semiconductor	It refers to a technology used to manufacture large-scale integrated circuit (IC) chips or chips made with this technology. It is a read-write RAM chip on the computer mainboard.
RO	Reverse Osmosis	Osmosis is adopted to put clear water and saltwater in one tube. The middle is separated by a semi-permeable membrane, allowing water to pass through. Then, water flows from the place with low osmotic pressure to the place with high osmotic pressure.
AIO	All In One	It supports synchronous and asynchronous (callback-based) processing.
EC	Erasure Coding	When the data block or check block is lost or damaged, the system can be recovered according to the EC algorithm to protect data.
SIFT	Scale-invariant feature transform	It can detect and describe the local features in the image. It looks for the extreme points in the spatial scale and extracts their position, scale, and rotation invariants.
IoS	Indicator of Success	It is mainly the latest indicator to evaluate the difference between the existing and optimized models.
DNA	Deoxyribonucleic acid	It is a molecule with a double-stranded structure and is made up of deoxyribonucleotides.
SQL	Structured Query Language	It is a special-purpose programming language for managing relational database management systems.
DDR3	Double-Data-Rate Three Synchronous Dynamic Random Access Memory	It aims at the next-generation memory technology of Intel's new chip.
AI	Artificial Intelligence	It refers to the system and machine that can imitate human intelligence to perform tasks and improve itself iteratively based on the collected information.

Zeng-yuan Chao was born in Haidong, Qinghai,P.R. China, in 1978. He received the bachelor‘s degree from Chengdu University of Information Technology, P.R. China. Now, he works in Weather Modification Center of Shijiazhuang Meteorological Bureau.

His research interest include intelligent equipment , cloud precipitation and weather modification. E-mail: czy93420@163.com

Received: July 17, 2022; Accepted: November 10, 2022.

Computer Science and Information Systems 20(2):657–678 https://doi.org/10.2298/CSIS220807013W

Predicting Smart Cities' Electricity Demands Using K-Means Clustering Algorithm in Smart Grid

Shurui Wang[1], Aifeng Song,[2,*] and Yufeng Qian[3]

[1] School of Electric and Electronic Engineering, Huazhong University of Science and Technology, Wuhan, 430000, China
[2] School of Management and Economics, North China University of Water Resources and Electric Power, Zhengzhou, China
saf0217@126.com
[3] School of Science, Hubei University of Technology, Wuhan, 430000, China

Abstract. This work aims to perform the unified management of various departments engaged in smart city construction by big data, establish a synthetic data collection and sharing system, and provide fast and convenient big data services for smart applications in various fields. A new electricity demand prediction model based on back propagation neural network (BPNN) is proposed for China's electricity industry according to the smart city's big data characteristics. This model integrates meteorological, geographic, demographic, corporate, and economic information to form a big intelligent database. Moreover, the K-means clustering algorithm mines and analyzes the data to optimize the power consumers' information. The BPNN model is used to extract features for prediction. Users with weak daily correlation obtained by the K-means clustering algorithm only input the historical load of adjacent moments into the BPNN model for prediction. Finally, the electricity market is evaluated by exploring the data correlation in-depth to verify the proposed model's effectiveness. The results indicate that the K-mean algorithm can significantly improve the segmentation accuracy of power consumers, with a maximum accuracy of 85.25% and average accuracy of 83.72%. The electricity consumption of different regions is separated, and the electricity consumption is classified. The electricity demand prediction model can enhance prediction accuracy, with an average error rate of 3.27%. The model's training significantly speeds up by adding the momentum factor, and the average error rate is 2.13%. Therefore, the electricity demand prediction model achieves high accuracy and training efficiency. The findings can provide a theoretical and practical foundation for electricity demand prediction, personalized marketing, and the development planning of the power industry.

Keywords: Smart city, smart grid, electricity prediction model, K-means clustering algorithm, back propagation neural network.

1. Introduction

With the rapid development of economic globalization, urbanization is accelerating. The problem of energy shortage is becoming even more obvious, such as water resource

* Correspondence author

lack, lack of non-renewable fossil resources, including coal and oil, and a serious waste of electricity. These increasingly prominent energy crises and environmental problems drive people to connect a considerable amount of distributed energy and energy storage devices. Meanwhile, green development and green economy have become the theme of economic development [1, 2]. The electric system is a critical element of urbanization, and intelligent electricity consumption is an inevitable factor in the construction of smart cities. The industrial society also puts forward higher reliability and quality requirements for power supply [3]. Consequently, scholars in related fields focus on applying intelligent prediction algorithms to intelligent electricity demand prediction.

The State Grid Corporation of China vigorously advocates the construction of a smart grid, and it leads the integration of various information technologies. It pursues the safety performance of the power system, the stability and distribution of power supplies, and the efficient utilization of energy, to reduce energy consumption and alleviate environmental pollution simultaneously in the power supply and distribution process [4]. The smart grid aims to optimize the production, distribution, and consumption of electric energy by obtaining more electricity consumption information from customers. Souri and Hosseini affirmed the business processing performance of big data technology in smart cities in their research [5]. The development of the Smart Grid makes accurate power load/demand prediction possible and increasingly urgent, helping power enterprises' scientific load dispatching and power production. Ultimately, the Smart Grid aims to achieve renewable energy in the power system. Introducing consumer decisions to reduce energy use will significantly reduce power load usage, which may reduce consumption in the coming years. Understanding the types of load prediction is helpful for the power system to manage the demand response plan effectively. Based on this, the load demand change can be predicted to calculate the required power generation and improve energy efficiency.

The electricity demand prediction for the smart grid is crucial for the power industry and critical to operation, planning, and control. Electricity demand prediction is a vital foundation for ensuring the safe operation of the power system, achieving scientific management and unified dispatching of the power grid, and formulating a reasonable development plan [6]. Electric energy consumption is affected by multiple factors. However, traditional electricity demand prediction methods take several factors as variables. It only considers the internal data of the power system, such as maximum load and regional average electricity consumption, ignoring numerous external factors [7]. Therefore, it is significantly crucial for the development of the power industry to investigate the electricity demand of the smart grid. Many high-resolution user load data can describe the user's power consumption habits and predict the user's power consumption. Therefore, mining user load data is of great value for grid system scheduling optimization, fine operation and management, and serving market users. According to the temporal and nonlinear characteristics of power load data, short-term load forecasting models can be divided into two categories. One is the time series approach, which usually regards the power load as a collection of time series. The prediction model is constructed according to the historical power load data and related influencing factors. In this way, the future load value can be predicted. The real-time load prediction method based on the user clustering strategy first takes a load of users at the hour as the feature. Then, it uses a clustering algorithm to classify users of different power consumption modes and a regression algorithm to forecast the load of classified

users. By superimposing the predicted load of various users, the total real-time predicted power load is obtained, and then the generation is predicted.

In summary, according to the new characteristics of the current development of smart cities and the new needs of the smart grid, it is of great practical value to predict the demand for smart electricity in the smart city. This work uses the K-means algorithm to cluster the historical load curves of users' electricity consumption. BPNN is used to construct different feature extraction network structures for different types of users, and the prediction model is trained. The contributions of this work are as follows. (1) A customer segmentation model is constructed based on traditional customer segmentation methods by analyzing the customer differences through the demographic, legal person, economic and geographic information provided by the Smart City Basic Database. As corporate reconstruction continues in China, the power industry is also experiencing such a change. The electric power companies with a fixed source of customers and stable economic benefits in China are previously owned by the government, and the power generation, power supply, and power distribution of the companies are under unified management by the national government. Nowadays, Chinese power companies have also been promoted to the market, which will face more competition. Thus, the power industry in China must comply with market rules. Meanwhile, private enterprises' practical power demands must be fully considered to strengthen the consciousness of serving customers and help them increase profits. (2) The relationship between the total electricity consumption and regional population, Gross Domestic Product (GDP), per capita GDP, the number of industrial enterprises above a specific size, and total imports and exports are analyzed by data from the Smart City Basic Database. An electricity demand prediction model is established by combining the internal data from power companies. Finally, its performance is analyzed through simulation to provide a reference for green energy use in subsequent smart cities.

2. Recent Related Work

There have been many studies on electricity demand prediction in the past few years. Classic and traditional electricity prediction methods have failed to adapt to the changing electricity market. Some new methods have been applied to electricity demand prediction. Mirjat et al. (2018) predicted that Pakistan's average electricity growth rate would be 8.35% in subsequent years using deep learning and electricity data from 2015 to 2017; this data was19 times the existing base [8]. Khalifa et al. (2019) used electricity consumption data to model the Qatar electricity market. They found that more energy consumption would be generated around 2030 and proposed to improve electricity efficiency by reducing electricity consumption [9]. Kim et al. adopted the Long Short-Term Memory (LSTM) algorithm of deep learning to predict electricity consumption and applied the one-hot coding method to the input and output values of electricity demand. This model had higher prediction accuracy than other general algorithms [10]. Then, the observed daily load curves were represented by a set of periodic smooth-spline basis functions. The basis function coefficients were obtained following the evolution of a linear Gaussian state space model. Nam et al. (2020) developed an energy prediction model by renewable energy technologies and implemented it in South

Korea's energy policy. They utilized the deep learning algorithm to predict fluctuations in electricity demand and power generation. Through experiments, they proved that this model achieved the lowest economic and environmental costs, generated stable electricity to meet demand, and realized the policy of 100% renewable energy [11]. The general LSTM hybrid model only optimizes the prediction front end. The optimization treatment of the residual at the back end of LSTM prediction is ignored, and the optimization treatment of the residual is missing. However, BPNN, as a multi-layer mapping network with forwarding information transmission and back error propagation, can achieve the purpose of error correction.

The above research shows that different models have been devised for electricity demand prediction and have achieved better model performance than traditional models in electricity demand prediction. However, some models only consider the demand of the electricity market without the influencing factors of electricity demand from other aspects during prediction. Therefore, it is necessary to determine the critical influencing factors using appropriate methods and massive data to establish a prediction model for electricity demand.

An electricity demand prediction model is proposed according to the actual situation by investigating literature about the smart city and smart grid. The model considers the new characteristics of current smart cities and the new needs of smart grids. A segmentation model is created by classifying electricity consumers and analyzing data from the smart city base database to study demographic, economic, and geographic differences. In addition, a new electricity quantity prediction algorithm is innovatively proposed, achieving high accuracy based on data correlation. The results can provide a theoretical and practical basis for smart grid construction. It provides new ideas for smart city construction.

3. Methods

3.1. Methods of electricity demand prediction

The three most commonly used electricity prediction methods are the classic prediction method, the traditional prediction model, and the intelligent prediction model. (1) Classic prediction methods include the trend extrapolation method, classified electricity demand prediction method, and load density method. Although these prediction methods are widely used, most analyze the relationship between some simple variables without deep data analysis. Thus, they cannot provide precise prediction results [12]. (2) Traditional prediction methods contain the regression analysis method, time series method, and random time series method. The regression analysis method establishes the relationship between the dependent variables and known load data. Then, it predicts the electricity system's load through mathematical analysis. By comparison, the time-series methods cover the exponential smoothing and the Census-H Decomposition methods. The random time series methods include the state space method, the Box-Jenkins method, and the Markov method [13]. According to the given data, the relationship

between the independent and dependent variables is determined, and the regression equations and various parameters are confirmed. Based on the given equation, the dependent variable is obtained from the existing independent variables, and finally, the electricity prediction data are obtained. (3) When there are large random factors in historical electricity demand, there may be errors in the prediction result caused by bad data in the time series. In recent years, the electricity market has become increasingly complicated. Classic prediction methods cannot adapt to the electricity market's nonlinear, multi-variable, time-varying, and random characteristics. Hence, some new prediction methods are used in electricity demand prediction. The laws are extracted to establish a knowledge base for reasoning and judgment based on real experience [14]. Table 1 illustrates the comparison results of the advantages and disadvantages of different prediction methods.

Table 1. Comparison of different electricity demand prediction methods

Recognition methods	Advantage	Disadvantage
Classic prediction method	The broadest range of applications and the most extended use time	Lack of scientific theory, and low prediction accuracy
Traditional predictive model	Substantial data analysis capabilities and high model accuracy	The algorithm runs for a long time and requires high system configuration
Intelligent prediction model	High prediction accuracy	Time-consuming database construction

3.2. K-Means Clustering (KMC) algorithm

(1) Algorithm utilization: at present, the construction of a smart grid must comply with market laws, and electric power enterprises must rely on its competitiveness for subsequent survival and development. It is crucial for enterprises to fully understand all customers, improve customer experience, and enhance customer loyalty. For customer segmentation, the traditional segmentation method uses a single indicator and cannot effectively divide customers. With the development of smart cities and the advancement of big data technology, a large amount of data can be obtained, while data mining technology can be used to extract the required indicators to segment power customers [15]. Currently, among various data mining algorithms, the K-Mean clustering algorithm has attracted the attention of many scholars due to its simple implementation and high efficiency.

The KMC algorithm uses the distance between two targets as an evaluation indicator to measure the similarity. When the distance between two objects is small, the similarity between the data is relatively high. This algorithm usually consists of relatively close objects. The final goal is to obtain a data group with a compact distance and a high degree of separation [16].

(2) Algorithm principle: the initial dataset is set to (x1, x2 … xn), and each data unit is a p-dimensional vector (the p-dimensional vector is composed of p eigenvalues). The KMC algorithm aims to divide the original dataset into K categories G= {G1, G2, …,

Gk} with a given number of categories k (k=n). Each iteration of the KMC algorithm must check whether the classification of each data unit is correct. The data must be adjusted if it is classified into the wrong category. The adjusted data is clustered with k points in the space as the center, and each cluster center's value is updated until the cluster center is a constant. The stable state of the cluster center indicates that the clustering criterion function has converged, and the best clustering result is obtained [17]. Figure 1 illustrates the implementation scheme of the KMC algorithm.

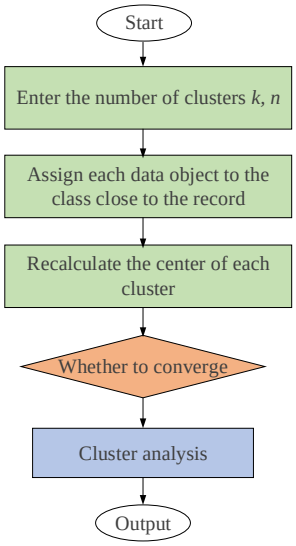


Fig. 1. The implementation scheme of the KMC algorithm

First, K data objects are arbitrarily selected from N data objects as initial clustering centers. The remaining objects are assigned to the cluster that is most similar to them (represented by the cluster center) according to their similarity (distance) to these cluster centers. Then, the cluster center of each new cluster (the mean of all objects in the cluster) is calculated, and the process is repeated until the criterion function begins to converge.

For consumer segmentation based on the KMC algorithm, the clustering result depends on the random selection of the initial cluster center. In practice, it has the advantages of simple description and easy implementation. It is scalable and efficient for processing large data sets. The specific implementation steps of the power consumer segmentation model are as follows. (1) K objects are randomly selected from n pieces of sample data as the initial cluster center. (2) The distance from each sample to each cluster center is calculated. The sample is assigned to the nearest cluster center category. (3) After all samples are allocated, it recalculates the centers of k clusters. (4) Compared with the k cluster centers obtained from the previous calculation, if the cluster center changes, turn to step (2); otherwise, turn to step (5). (5) When the center does not change, the algorithm flow stops and outputs the clustering results. Before any operation, the user load data in the prediction area and is preprocessed to identify and

correct bad data. According to the different correlations of its time series, the user load series of different categories are converted into tensors and then input into BPNN for abstract feature extraction. Furthermore, according to the differences in total and industry load characteristics, the k-means algorithm is used to aggregate load curves to obtain industry-typical load curves, and the user clustering curves are further extracted to form group loads. Finally, the group load is predicted by selecting appropriate monthly and annual forecasting methods.

3.3. Back propagation neural network (BPNN) algorithm

The Artificial Neural Network (ANN) is composed of numerous interconnected neurons and has a strong nonlinear mapping ability [18]. The BPNN is a multi-layer feedforward network trained according to the backpropagation algorithm [19]. The topological structure of the BPNN includes an input layer, a hidden layer, and an output layer, as shown in Figure 2.

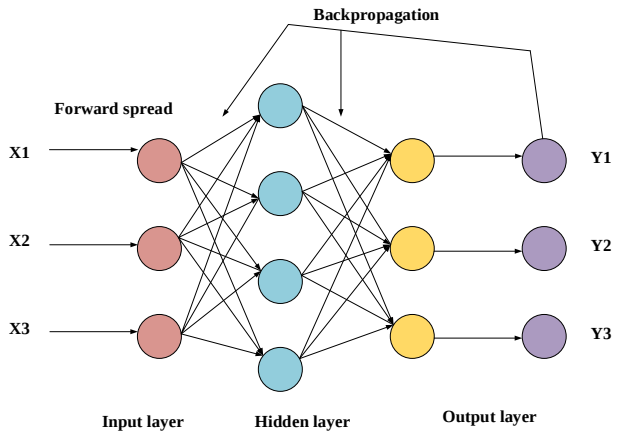


Fig. 2. Structure of BPNN

The BPNN is usually composed of multiple layers and multiple neurons, generally including an input layer, a hidden layer, and an output layer [20]. The specific flow of the BPNN prediction model is shown in Figure 3. The input vector can be expressed as:

$$x = \left[x_1, x_2, x_3 \ldots x_i, \ldots x_m\right], i = 1, 2, \ldots . m \tag{1}$$

The output vector can be written as:

$$y = \left[y_1, y_2, y_3 \ldots y_k, \ldots y_n\right], k = 1, 2, \ldots . n \tag{2}$$

Equation (3) describes the neuron input of the hidden layer.

$$h^{(l)} = \begin{bmatrix} h^{(l)}{}_1, h^{(l)}{}_2, h^{(l)}{}_3 \\ \ldots h^{(l)}{}_j, \ldots h^{(l)}{}_{sl} \end{bmatrix}, j = 1, 2, \ldots . sl \tag{3}$$

In Equation (3), *sl* denotes the number of neurons in the *l*-th layer. Assumed that $w^{(l)}{}_{ij}$ represents the connection weight associated with the *j*-th neuron in the *l*-th layer, $b^{(l)}{}_i$ refers to the threshold of the *i*-th neuron in the *l*-th layer, and $net^{(l)}{}_i$ stands for the input of the *i*-th neuron in the *l*-th layer. Then:

$$h^{(l)}{}_i = f(net^{(l)}{}_i) \tag{4}$$

$$net^{(l)}{}_i = \sum_{j=1}^{sl-1} w^{(l)}{}_{ij} h^{(l\text{-}1)}{}_j + b^{(l)}{}_i \tag{5}$$

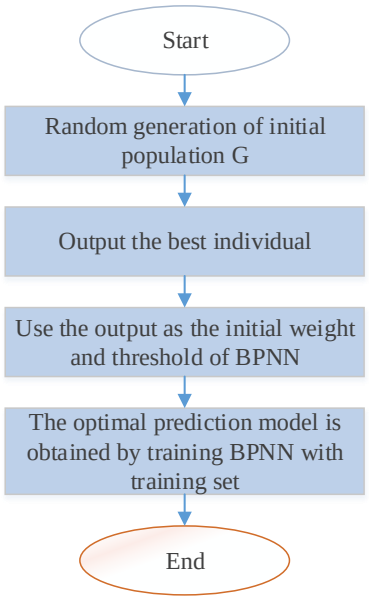


Fig. 3. Specific flow of BPNN prediction model

3.4. Power consumer segmentation model

(1) Overall framework: based on the above theory, the Smart City Basic Database is utilized to establish a functional structure model for power consumer segmentation (Figure 4). First, a data warehouse is established. Then, relevant customer segmentation

data is extracted for data analysis. Moreover, data is cleaned and conversed. The association analysis method is adopted for data mining, and finally, the mining results are analyzed. The Smart City Basic Database is the foundation of the entire model. Pre-processing of data is the guarantee for real and effective mining results. The effectiveness of customer segmentation depends largely on selecting customer consideration standards and establishing measurements. The adopted mining method based on actual needs is the key to the entire model.

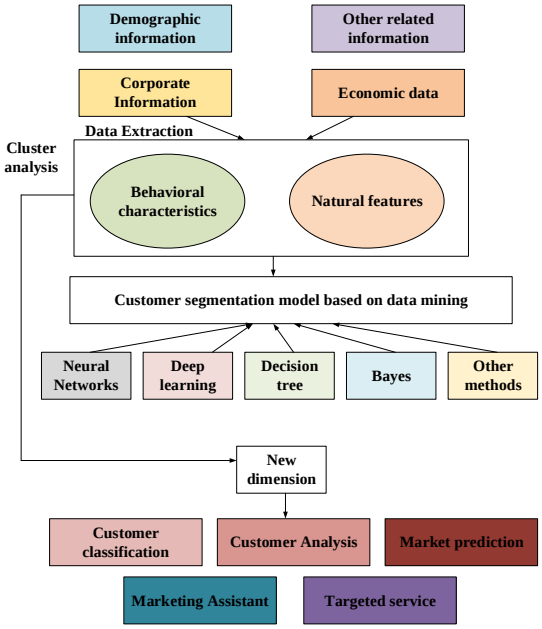


Fig. 4. Functional structure of power consumer segmentation

(2) Customer segmentation: Currently, electricity data analysis research and technology use traditional data extraction methods and statistics. The Smart City Basic Database's data is utilized to analyze electricity data to optimize the accuracy and usability of the algorithm. Besides, the big data method is utilized for prediction to maximize the type of experimental data. The previously impossible prediction task can be completed by exploring the data association relationship at present, which can ensure high precision simultaneously. The population electricity value information directly reflects the individual electricity value of customers. It indirectly demonstrates the gathering area of high-potential customers through personal information and social insurance information. In load (electricity demand) prediction, users are subdivided and predicted separately. That is, the major categories of power consumption characteristics are understood. Then, a regression algorithm is used to model and predict each cluster's load and add up each cluster's prediction results to form the final urban load forecast. Additionally, the prediction results can be compared with the actual historical data to generate an evaluation. The evaluation

will be fed back to the prediction model. The accuracy of the prediction model can be improved by adjusting the modeling parameters accordingly.

(3) Data processing: first, the Dongfangtong TI-ETL tool and data desensitization technology are utilized to transform sensitive or confidential information to protect private data. Some missing data, including names, gender, and address of customers, is extracted from other information. Various social insurance databases are integrated, and the collected information is used to roughly restore the demographic information and provide the basis for power consumer segmentation.

(4) Algorithm realization: it is divided into population electricity value information, enterprise commercial value distribution, and macroeconomic information. Figure 5 reveals the population electricity value information. The power consumers are segmented. The resident data is arranged in a table from high to low in the order of individual units according to residents' social insurance information, social security information, corporate information, and the potential and influence of electricity use. Each administrative district is taken as a unit. As for the data of the corporate legal person, the legal person's registered capital is used as the analysis target, and the social insurance information is determined according to the insurance amount. In addition, the social security information depends on the subsidy amount, and the housing provident fund is based on the monthly payment amount [21].

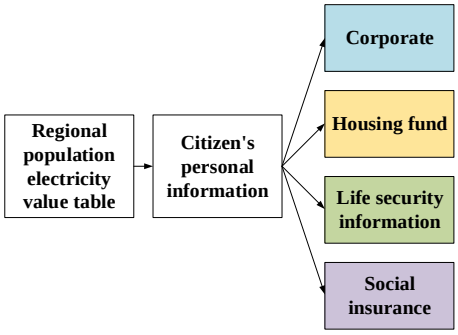


Fig.5. Structure of population electricity value realization

The enterprise commercial value distribution is presented in Figure 6. Obviously, the organization code is used as the search basis to match the source data from each commission, office, and bureau. The evaluation and ranking are based on four dimensions: business category, registered capital, annual turnover, and the number of employees. Each administrative district is taken as a unit. For corporate legal person's data, KMC analysis is performed based on turnover, registered capital, and the number of employees. Among various business categories, the conversion weight of turnover for the construction industry is 10%, that for the manufacturing industry is 100%, that for the wholesale and retail industry is 30%, and that for the service industry is 15% [22].

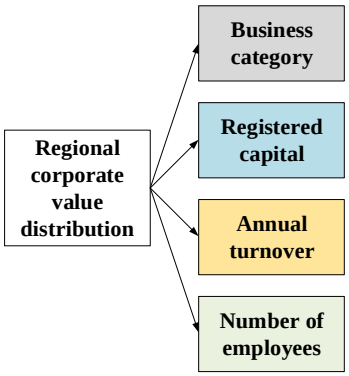


Fig. 6. Structure of enterprise commercial value realization

In terms of macroeconomic information, the macro value evaluation aims to evaluate the administrative districts' electricity consumption potential to segment power consumers. The data from some districts are accurate to the administrative streets. Previous studies have selected several significant data categories, such as regional GDP, per capita disposable income, per capita GDP, total asset investment, and trade data [23].

In a city, five districts, A, B, C, D, and E, are chosen for empirical research. In District A, four companies are chosen from the industrial area A11-A14. In District B, four companies are chosen from the office building areas B11-B14. In District C, four neighborhoods are selected from the residential areas C11-C14. District D and District E are comparison controls.

3.5. Model of electricity demand prediction

(1) Influencing indicators: electric energy consumption is affected by multiple factors. However, traditional prediction methods only consider a few factors as variables. These methods only consider the internal data of electricity companies and fail to fully consider the impact of changes in other factors on prediction, ultimately providing limited prediction accuracy [24]. The external information of electricity companies provided by the Smart City Basic Database is fully utilized. Given the impact of changes in various relevant objects on the prediction value, the regional annual electricity consumption is predicted considering multiple influencing factors.

In addition to the electricity companies' factors, the total electricity consumption is also affected by many factors such as population, corporate trends, economic conditions, energy policies, and electricity price adjustments. In particular, the development of the social economy needs to consume a large amount of electric energy, and there is a correlation between electricity consumption and economic indicators [25]. There are many statistical dimensions of financial data. The district's social product, per capita GDP, price index, total import and export, and other economic indicators of the electricity companies are chosen to establish an electricity consumption prediction model [26]. The specific names of the data indicators are as follows: (1) macroeconomic

data; (2) policies and other external data; (3) regional electricity consumption data in past years; (4) power consumer segmentation data.

(2) Normalization processing: load prediction is still very challenging because it depends on external factors such as weather and exogenous variables. The model's characteristics (including the prediction period or the variables used) are reflected in the structure of the neural network. The empirical selection method is used to determine the model input variables, and then the power load prediction model is established. ANN usually normalizes the data before training. The training effects of different transfer functions are different to avoid neuron oversaturation [27]. The input data value must be within [0,1], which is the characteristic requirement of the transfer activation function. Therefore, the original data of the network must be processed. The original data are normalized, and the equation is as follows:

$$x_{i0} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \tag{6}$$

In Equation (6), x_{i0} denotes the *i*-th feature parameter after normalization, x_i represents the original *i*-th feature parameter, $x_{\min}$ signifies the minimum value of the *i*-th feature parameter, and $x_{\max}$ stands for the maximum value of the *i*-th feature parameter.

(3) Hyperparameter setting: after preliminary experiments, a three-layer network model structure with a hidden layer is determined. The number of neurons is 18, and the logsig transformation function is used. The number of neurons in the second layer is the same as the number of output variable vectors, and the output layer uses a pure linear transformation function. The input feature parameters are 65; that is, the number of input layer nodes is 65, and the number of output layer nodes is 5. Generally, increasing the number of nodes in the hidden layer can reduce the network's training error rather than increasing the number of hidden layers. The BPNN algorithm can be set as a three-layer structure to map the n-dimensional input layer to the *m*-dimensional output layer. Therefore, the number of hidden layers in the network is determined as 1. When applying a neural network for electricity prediction, the reference equations for selecting the number of hidden layer neurons are as follows:

$$h = 2m + 1 \tag{7}$$

$$h = \sqrt{n+m} + \alpha * \tag{8}$$

In Equations (7) and (8), *h* represents the number of nodes in the hidden layer, *m* denotes the number of nodes in the input layer, and *n* refers to the number of nodes in the output layer. After a comprehensive comparison of experiments, the number of nodes in the hidden layer is selected as 18. The S-tangent tansing is selected as the activation function for hidden layer neurons, and the activation function for output layer neurons is the S-type logarithmic logsig function.

(4) Data source: the streaming data in the power grid is collected from smart meters, PMUs, and various sensors. The data is extensive in scale, diverse in structure, and fast in speed. The electric power company has installed many smart meters to accurately obtain customers' electricity consumption data of different electric equipment. The meters will send real-time electricity consumption information to the grid every 5 minutes. Consequently, streaming data collection requires fast collection speed, high reliability, real-time monitoring of data changes, and simple data processing. Therefore, the collection system is a distributed, reliable, and highly available system of massive log aggregation, which can monitor and receive data from the client and send it out. The log file is transferred to other nodes without loss when a node fails, ensuring data integrity. The collected data is divided into a training set and a test set by the ratio of 8:2, and the percentage of each data type in the two datasets is consistent.

3.6. System improvement and verification

(1) System improvement: the influence of changing trends on the error surface. The BPNN may fall to a local minimum, which the additional momentum can prevent. The adjustment equations for weight and threshold with additional momentum factor are:

$$\Delta w_{ij}(k+1) = (1-mc)\eta\sigma_i p_j + mc\Delta w_{ij}(k) \tag{9}$$

$$\Delta b_{ij}(k+1) = (1-mc)\eta\sigma_i p_j + mc\Delta b_{ij}(k) \tag{10}$$

In Equations (9) and (10), *w* denotes the weight vector, *k* refers to the number of trainings, and *mc* represents the momentum factor. In addition, η stands for the learning rate, Δb_{ij} signifies the gradient of the error function, and σ_i and p_j are the correlation coefficients.

Power load prediction is susceptible to many elements. These influencing factors are used as independent variables to explain the changes in dependent variables. When there is a linear relationship between multiple dependent and independent variables, the regression problem becomes a multiple linear regression. The prediction model is as follows:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n + \varepsilon \tag{11}$$

In Eq. (11), β_0 is a constant term. $\beta_1, \beta_2, ..., \beta_n$ represents a regression coefficient. $x_1, x_2, ..., x_n$ refers to the n influencing factors related to the load. Y stands for the predicted value, and ε indicates a random disturbance term.

(2) Clustering algorithm evaluation: here, the KMC algorithm uses the square error and criterion function to evaluate the clustering performance. *X* represents the given dataset, and each data unit is a *p*-dimensional feature vector. It is set to *K* categories. The algorithm randomly selects *k* data as the starting cluster center to analyze the distance from each data unit to the cluster and divides the data into the array sink where the

corresponding cluster center is located. It is supposed that X contains k data groups X_1, X_2 … X_k, the amount of data units in each data group is m_1, m_2, ..., m_k; the cluster centers of each data group are n_1, n_2, ..., n_k. The used square error equation [28] is:

$$E = \sum \sum \left\| p - m_i \right\| \tag{12}$$

(3) Training of BPNN: during the training, there are 18 neurons in the hidden layer of the BPNN network with the logsig transform function. The number of neurons in the second layer is the same as the number of output variable vectors, and the output layer adopts the pure linear transform function. The selection of expected training errors is related to the number of neurons and the training time. The minimum error boundary E of the general neural network ranges from 0.001 to 0.1. When the error value of the cyclic calculation is smaller than E, the calculation is stopped, and the calculation results are output. Otherwise, the calculation continues until the error meets the design requirements. Here, E is set to 0.02. In the selection of the learning rate, the excessively large learning rate will affect network stability. In contrast, the extremely small learning rate will prolong the learning time of the network, and convergence will deteriorate. Therefore, the appropriate learning rate should be selected according to the specific network model structure. Here, the learning rate is set to 0.05, and the network will be trained 10,000 times. The preprocessed data matrix is imported into MATLAB and normalized. The Newff function is employed to establish a BPNN model.

The data for the experiment comes from the London household electricity Usage Record (UK-DALE) provided by the UK Energy Centre. The dataset is collected using a smart meter. The original power demand data are household load data collected with 6s as the acquisition frequency. Then, the model forecasts 1-hour household electricity demand. Therefore, the data unit is converted first, and the load data is converted to the electricity consumption within 6 seconds, and then the data is merged. 90% of the combined data are selected as the training set, and 5% are used as the validation set. 10% is used as the test set. After the calculation, denormalization processing is performed on the data. The MATLAB toolbox calculates the relative error percentage and the predicted electricity demand value.

4. Experimental Results

4.1. Results of power consumer segmentation

Different electricity consumption areas can be divided through the above customer segmentation model, as shown in Table 2 and Figure 7. The four companies in District A are in the high electricity consumption area in their administrative district. The medium electricity consumption area is the office area of District B. The area with low electricity consumption is the residential electricity area of District C. This is consistent with the actual result. Then, all areas are divided into residential electricity and commercial electricity areas. According to the model, the number of all residential

customers and commercial customers in the city can be subdivided into four levels from P1-P4, from more to less electricity consumption. The number of residents with the highest electricity consumption reached 25,678. The number of commercial customers with the highest electricity consumption reached 298. Such results also confirm the effectiveness of the segmentation model proposed here.

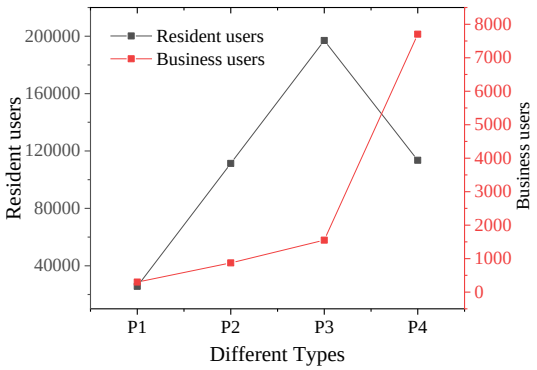


Fig. 7. Performance analysis of the model based on BPNN

Table 2. Regional electricity potential distribution

Area category	High electricity consumption area	Middle electricity area	Low electricity area
Bend over	A11	B11	C11
	A12	B12	C12
Squat down	A13	B13	C13
	A14	B14	C14

Each feature clustering's center value is analyzed for clustering user groups. According to the difference in performance features of user categories, the power user value is grouped to formulate the most reasonable value response strategy. The strategy can optimize the power sales side. The prototype system of power user payment data is designed and developed to visualize the clustering results of power users. The system can characterize the users of power enterprises.

4.2. Performance analysis of the electricity demand prediction model

Figure 8 shows the results of electricity demand prediction calculated by the BPNN model. Figure 9 shows the model's performance analysis results based on BPNN. The electricity demand prediction results of different areas are close to the actual value, and the largest relative error is 5.129%. The minimum value is 2.294%, and the mean relative error (MRE) is 3.2671%. Hence, the BPNN has a good prediction effect. As shown in Figure 8, the model's training error is analyzed, and the results suggest that the

model tends to be stable when it is trained for 5,000 s. In contrast, the training speed of the algorithm of related research is excessively long, failing to meet the actual demand.

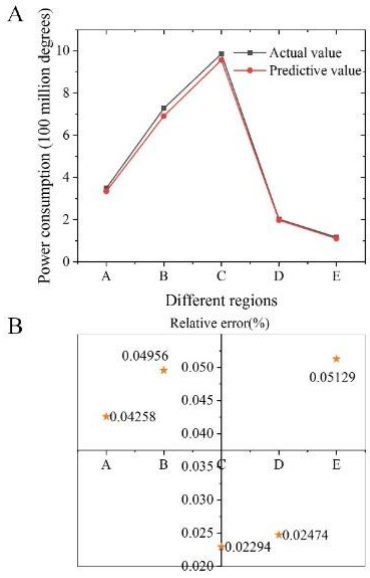


Fig. 8. Electricity demand prediction results based on BPNN

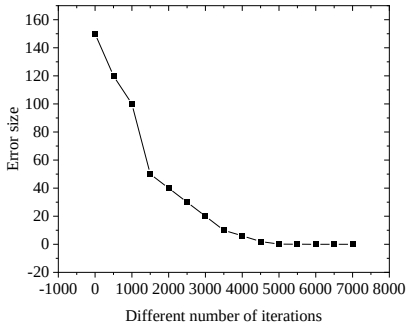


Fig. 9. Performance analysis of the model based on BPNN

4.3. Performance analysis results of the improved algorithm

After optimization, according to the results shown in Figures 10 and 11, the improved BPNN algorithm has a smaller MRE, with an MRE of 2.13%, and a faster training speed

compared with the unimproved BPNN. Besides, the improved algorithm reached a stable state in the 2000s, and its training accuracy is higher, which is more advantageous in predicting electricity demand. Hence, the improved algorithm meets the basic requirements of the electricity company for electricity prediction and has particular practical application value.

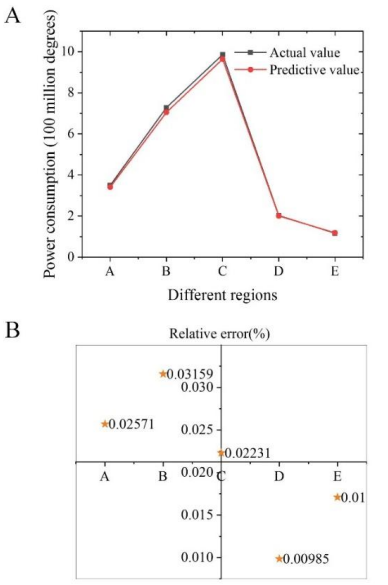


Fig. 10. One kernel a electricity demand prediction results based on improved BPNN

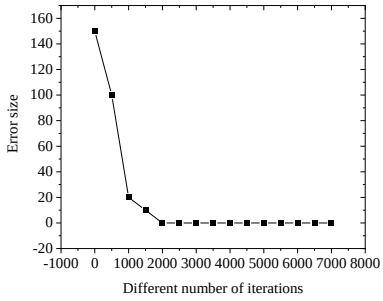


Fig. 11. Model performance analysis based on improved BPNN

4.4. Performance comparison of different models

Table 3 summarizes the comparison results of the model reported here and the latest models. The accuracy of the BPNN reported in the literature [29] is relatively high and remains at 79.63%. Compared with methods in other literature, the proposed model has the best performance. The accuracy rate of the model is 85.25%, and the average value is also the largest, 83.72%. Hence, the proposed model is significantly better than the latest models in terms of algorithm performance. Accuracy (%)

Table 3. Performance comparison results of different models

Models	Accuracy (%)	Recall (%)	F1-value	Mean value
Literature [29]	0.7963	0.6625	0.8253	0.761366667
Literature [30]	0.8142	0.6768	0.8649	0.7853
Literature [31]	0.7125	0.6972	0.8165	0.742066667
Literature [32]	0.8073	0.7323	0.8246	0.788066667
The proposed model	0.8525	0.7848	0.8745	0.837266667

5. Discussion and Analysis

The smart city system's comprehensive application of big data can provide trend judgments and information sharing. Doing so promotes the innovative management and interconnection of cities, the healthy competition of various industries, and the sustainable development of society. Besides, with the increasingly prominent energy crisis and environmental problems, many distributed energy sources and energy storage devices are connected, and industrial society has put higher requirements for power reliability and quality. The load forecast is of great significance to the dispatch of the power system, the fundamental basis of formulating power generation and transmission plans, and an influential aspect of the modern development of the power market. Improving the accuracy of power system load forecasting can effectively improve the power sector's economic benefits and promote the power grid's safe and economical operation. Here, the demand forecasting of the power system is studied by consulting many relevant domestic and foreign documents.

The characteristics and significance of power system demand forecasting are considered under the big data from smart cities. Traditional power data analysis methods rely primarily on limited sample data. New algorithms are proposed and improved based on proprietary data in this industry and field to achieve higher prediction accuracy and faster processing speed. The big data method for forecasting requires as many types of data as possible. The previously impossible forecast task is completed by exploring the data association relationship, ensuring high accuracy. During power information forecasting, the focus of work is shifted from researching complex algorithms to preparing big data acquisition. On this basis, the existing methods of electricity demand prediction are compared, and their characteristics are analyzed, which is also reported in

the relevant literature [33]. Here, the data mining analysis is performed on the constructed data set, and the KMC algorithm is employed to establish a power customer segmentation model. After setting the specific hyperparameters of the model reported here, the initial data on the smart city is processed deeply. Finally, the validity of the customer segmentation model is verified, and the errors are quantitatively analyzed. The MRE is 3.2671 %, indicating that the model system is stable. Through cleaning, conversion, and establishment of new data warehouses, its intrinsic data characteristics are pointed out, and the intrinsic value of data is tapped [34]. Through the experiment, the model's accuracy reaches 85.25%, with the biggest mean of 83.72%. The MRE remains below 5%. Apparently, unlike traditional methods, the proposed customer segmentation model apparently depends on the smart cities' big data. It can describe customer behavior more comprehensively and accurately. Furthermore, it provides a reference for power companies to formulate appropriate marketing strategies and other in-depth research.

6. Conclusion

A smart grid customer segmentation model is constructed based on the KMC algorithm for electricity demand prediction by establishing a database and mining related data through big data analysis. The intrinsic value of data is deeply explored through data cleaning, data conversion, and establishing new data warehouses. Compared with the traditional customer segmentation method, the proposed model can describe customer behavior more comprehensively and accurately, relying on the Smart City Basic Database. In addition, the results of power consumer segmentation are incorporated into the input samples. The BPNN method is utilized for electricity demand prediction on the electricity system to establish a BPNN model. Finally, the input data is normalized. According to the simulation result, the accuracy rate of the constructed model algorithm reaches 85.25%, and the error value is also below 5%, which verifies the model's validity. This model can provide a reference for future green use of power energy in smart cities. The subdivision of the power consumers' user groups can help understand the demand of the power side and then predict the power generation. Such a division is of great value for the planned power generation of the Smart Grid and for improving the user service experience and satisfaction.

Although a suitable power demand forecasting model has been established in this study, there are some deficiencies. Firstly, there is no mature multi-index forecasting model for the relationship between time, social factors, and electricity demand. In addition, the constructed power demand forecasting model is still in the research stage based on actual cases, which lacks general applicability. Secondly, it is necessary to model the impact of annual weather conditions on power load. In power load prediction, electricity consumption has been accurately analyzed according to weather changes. However, the weather changes are macroscopic within a year, and it is difficult to measure them accurately in local areas. Therefore, in follow-up studies, it is worth striving to generate accurate digital reports on economic activities, people's lives, trade, and transport in specific regions. In addition, with the deepening development of the power market, the electricity demand prediction will show different characteristics.

Building suitable and precise prediction models for different business needs is also essential.

References

1. Rout, S. S., *et al.*, "Smart water solution for monitoring of water usage based on weather condition," International journal, Vol. 8, No. 9, 1-9. (2020)
2. Alhazmi, M., *et al.*, "Optimal integration of interconnected water and electricity networks," energy, 2021,4: 6. Vol. 4, 6. (2021)
3. Zhang K., *et al.*, "Security and privacy in smart city applications: Challenges and solutions," IEEE Communications Magazine, Vol. 55, No. 1, 122-129 (2017)
4. Anthopoulos, L., "Smart utopia VS smart reality: Learning by experience from 10 smart city cases," Cities, Vol. 63, No. 2, 128-148. (2017)
5. Souri, A. and Hosseini, R. "A state-of-the-art survey of malware detection approaches using data mining techniques," Human-centric Computing and Information Sciences, Vol. 8, No. 1, 1-22. (2018)
6. Mengelkamp, E., *et al.*, "A blockchain-based smart grid: towards sustainable local energy markets," Computer Science-Research and Development, Vol. 33, No. 1-2, 207-214. (2018)
7. Ahmad, T. and Chen, H. "Potential of three variant machine-learning models for forecasting district level medium-term and long-term energy demand in smart grid environment," Energy, Vol. 160, 1008-1020. (2018)
8. Mirjat, N. H., *et al.*, "Long-term electricity demand forecast and supply side scenarios for Pakistan (2015–2050): A LEAP model application for policy analysis," Energy, Vol. 165, 512-526. (2018)
9. Khalifa, A., Caporin, M. and Di Fonzo, T. "Scenario-based forecast for the electricity demand in Qatar and the role of energy efficiency improvements," Energy Policy. Vol. 17, No. 2, 155-164. (2019)
10. Kim, K. H., *et al.*, "Deep Learning Based Short-Term Electric Load Forecasting Models using One-Hot Encoding," Journal of IKEEE, Vol. 23, No. 3, 852-857. (2019)
11. Nam, K., *et al.*, "A deep learning-based forecasting model for renewable energy scenarios to guide sustainable energy policy: A case study of Korea," Renewable and Sustainable Energy Reviews, Vol. 122, 109725. (2020)
12. Kim, M., *et al.*, "A hybrid neural network model for power demand forecasting," Energies, Vol. 12, No. 5, 931. (2019)
13. Ghalehkhondabi, I., *et al.*, "An overview of energy demand forecasting methods published in 2005–2015," Energy Systems, Vol. 8, No. 2, 411-447. (2017)
14. Boroojeni, K. G., *et al.*, "A novel multi-time-scale modeling for electric power demand forecasting: From short-term to the medium-term horizon," Electric Power Systems Research, Vol. 142, 58-73. (2017)
15. Azaza, M. and Wallin, F. "Smart meter data clustering using consumption indicators: responsibility factor and consumption variability," Energy Procedia, Vol. 142, 2236-2242. (2017)
16. Bai, L. *et al.*, "Fast density clustering strategies based on the k-means algorithm," Pattern Recognition, Vol. 71, 375-386. (2017)
17. Zhang, G., Zhang, C., and Zhang, H. "Improved K-means algorithm based on density Canopy," Knowledge-based systems, Vol. 145, 289-297. (2018)
18. Whittington, J. C and Bogacz, R. "Theories of error back-propagation in the brain," Trends in cognitive sciences. 2019;23(3):235-250. Vol. 23, No. 3, 235-250. (2019)
19. Neftci, E. O. *et al.*, "Event-driven random back-propagation: Enabling neuromorphic deep learning machines," Frontiers in neuroscience, Vol. 11, 324-331. (2017)

20. Scellier, B. and Bengio, Y. "Equilibrium propagation: Bridging the gap between energy-based models and backpropagation," Frontiers in computational neuroscience. 2017;11:24-31. Vol. 11, 24-31. (2017)
21. Kim, H. W. and Jeong, Y. S. "Secure authentication-management human-centric scheme for trusting personal resource information on mobile cloud computing with blockchain," Human-centric Computing and Information Sciences, Vol. 8, No. 1, 11-23. (2018)
22. Li, F. *et al.*, "A clustering network-based approach to service composition in cloud manufacturing," International Journal of Computer Integrated Manufacturing, Vol. 30, No. 12, 1331-1342. (2017)
23. Della Peruta, M. "Adoption of mobile money and financial inclusion: a macroeconomic approach through cluster analysis," Economics of Innovation and New Technology, Vol. 27, No. 2, 154-173. (2018)
24. Olsson, T., Barcellos, L. F. and Alfredsson, L. "Interactions between genetic, lifestyle and environmental risk factors for multiple sclerosis," Nature Reviews Neurology, Vol. 13, No. 1, 25-33. (2017)
25. Gupta, V. and Pal, S., "An overview of different types of load forecasting methods and the factors affecting the load forecasting," International Journal for Research in Applied Science & Engineering Technology (IJRASET), Vol. 5, No. 4, 729-733. (2017)
26. Amara, F. *et al.*, "Household electricity demand forecasting using adaptive conditional density estimation," Energy and Buildings, Vol. 156, 271-280. (2017)
27. Pořízka, P. *et al.*, "Impact of laser-induced breakdown spectroscopy data normalization on multivariate classification accuracy," Journal of Analytical Atomic Spectrometry, Vol. 32, No. 2, 277-288. (2017)
28. Satapathy, S. C. *et al.*, "Multi-level image thresholding using Otsu and chaotic bat algorithm," Neural Computing and Applications. Vol. 29, No. 12, 1285-1307. (2018)
29. Wang, D. *et al*., "Multi-step ahead electricity price forecasting using a hybrid model based on two-layer decomposition technique and BP neural network optimized by firefly algorithm," Applied Energy, Vol. 190, 390-407. (2017)
30. Casteleiro Roca, J. L. *et al.,* "Short-term energy demand forecast in hotels using hybrid intelligent modeling," Sensors, Vol. 19, No. 11, 2485-2496. (2019)
31. Dinesh, C. *et al.,* "Residential power forecasting using load identification and graph spectral clustering," IEEE Transactions on Circuits and Systems II: Express Briefs, Vol. 66, No. 11, 1900-1904. (2019)
32. Johansson, C. *et al.,* "Operational demand forecasting in district heating systems using ensembles of online machine learning algorithms," Energy Procedia, Vol. 116, 208-216. (2017)
33. Al Ogaili, A. S. *et al*., "Review on scheduling, clustering, and forecasting strategies for controlling electric vehicle charging: challenges and recommendations." Ieee Access, Vol. 7, 128353-128371. (2019)
34. Kazemzadeh, M. R. *et al*., "A hybrid data mining driven algorithm for long term electric peak load and energy demand forecasting," Energy, Vol. 204, 117948-117956. (2020)

Shurui Wang was born in Tangshan, Hebei, P.R. China, in 1999. He is studying for an undergraduate degree at Huazhong University of Science and Technology, P.R. China. His research interest include electric power system.

Song Aifeng, Ph.D., lecturer, North China University of Water Resources and Electric Power, research direction: decision-making theory and method, energy efficiency, sustainable supply chain.

Yufeng Qian was born in Wuhan, Hubei, China, in 1986. He received the B.S. degree in information and computing science from Central China Normal University, Wuhan, China, the M.S. degree and the Ph.D. degree in computational mathematics from Wuhan University, Wuhan, China, in 2008, 2010 and 2013, respectively. Currently, he works as a lecturer in the School of Science, Hubei University of Technology, Wuhan, China. His research interests include applied mathematics, complex systems and complex networks.

Received: August 07, 2022; Accepted: December 02, 2022.

Computer Science and Information Systems 20(2):679–700 https://doi.org/10.2298/CSIS220813014Y

Using Machine Learning Approach to Construct the People Flow Tracking System for Smart Cities

Baofeng Yao*, Shijun Liu, and Lei Wang

College of Computer Engineering, Bengbu University,
Bengbu 233000, Anhui, China
ybf@bbc.edu.cn
liushijun@bbc.edu.cn
bbxywl@bbc.edu.cn

Abstract: In the crowd congestion in smart cities, the people flow statistics is necessary in public areas to reasonably control people flow. The You Only Look Once-v3 (YOLOv3) algorithm is employed for pedestrian detection, and the Smooth_L1 loss function is introduced to update the backpropagation parameters to ensure the stability of the object detection model. After the pedestrian is detected, tracking the pedestrian for a certain time is necessary to count out the specific number of pedestrians entering and leaving. Specifically, the Mean Shift algorithm is combined with the Kalman filter to track the target. When the target is lost, the Mean Shift algorithm is used for iterative tracking, and then the Kalman prediction is updated. In the experiment, 7,000 original images are collected from the library, mentioning 88 people of which 82 are recognized, and the detection accuracy reaches 93.18%. The 12,200 original images collected in the teaching building include149 people, of which 139 are recognized, with the detection accuracy reaching 93.29%. Therefore, the people flow statistics system based on machine vision and deep learning can detect and track pedestrians effectively, which is of great significance for the people flow statistics in public areas in smart cities and for the smooth development of various activities.

Keywords: smart city; machine learning; machine vision; people flow tracking system; YOLOv3

1 Introduction

At present, construction of smart cities is intensively pursued worldwide as a result of the continuous growth of the population, and the corresponding leading technologies are applied in Europe, North America, Japan, and South Korea. The construction of the smart cities begins in first-tier cities and developed in second-tier cities, and the increasingly "smart" cities bring huge impacts on people's lives [1]. While improving the quality of lives of people, the "smart" city sees increased people flow compared to the past because it brings about complex and diverse "smart" activities. Especially in stations, airports, scenic spots, and large shopping malls, the people gathering results in queuing, congestion, and even trampling in public areas, bringing many security risks to

* Corresponding author

people's social activities [2,3]. Therefore, in public places, timely diversion of the crowd is urgently to be solved. In the process, real-time statistics of the number of people in the area become very important. Through the statistics of people flow, the people flow density can be strictly controlled to maintain social order and ensure the safety of people in public areas [4].

In recent years, the regional population statistics and people flow density analysis have attracted widespread attention in many disciplines such as big data, smart city, and public services [5]. With the rapid advancement of technologies such as artificial intelligence, computer vision, and deep learning, people flow statistics are the basis for the detection of target pedestrians and the rational allocation of resources. Surveillance cameras are widely deployed and people flow statistics are extensively applied in smart cities. With the emergence of the people flow statistics based on machine vision, the pedestrian detection and multi-target tracking become the research hot spots [6,7]. Pedestrian detection is to segment the pedestrian area in the surveillance video, and then track the detected pedestrian target and identify its movement direction, providing data support for the statistics and prediction of the people flow in the area. Currently, the machine learning is applied to integrate the pedestrian features to find the basic characteristics of moving targets, and then deep learning-related algorithms are employed to classify the pedestrian features. The detection performance of the model established is subsequently tested [8-10]. However, the detection of moving pedestrians is still shortcoming due to the dynamic variability of the actual environment and the irregularity of pedestrian movement. Ullah et al. (2022) [11] proposed an intelligent and efficient human flow anomaly detection and recognition system based on artificial intelligence and big data technology. The self-pruning fine-tuned lightweight convolutional neural network classifies the ongoing events in the AI IoT environment as normal or abnormal. When abnormal human traffic is detected, the edge device alerts the relevant department and transmits the abnormal frame to the cloud analysis center. Dong et al. (2021) [12] designed an Ethernet-based flow detection and alarm system based on artificial intelligence. If an exception occurs (such as a long retention time), the system will alert the server. The workflow of parameter setting system and alarm system based on Web Server is introduced. The system can monitor and discover abnormal information in real time, interact with remote web server through Ethernet, and monitor special area effectively.

Statistics of human flow under intelligent video surveillance can be applied in commercial fields and play an important role in security and transportation. The people flow statistics is important to prevent dangerous events caused by excessive crowd density in public places. Therefore, a people flow statistics system is designed for smart cities based on machine vision and deep learning. In the study, the development status of comprehensive urban body under smart cities is analyzed first, and human target recognition and target tracking technology are then discussed to relieve the pressure of public facilities. The pedestrian detection in this study is implemented by YOLOv3 algorithm, and the Smooth_L1 loss function is innovatively introduced aiming at the model instability caused by L2 loss function in the process of network backpropagation parameter update. The tracking algorithm of Kalman filter is simulated according to its basic principle, and an improved tracking algorithm using Mean Shift is proposed. Finally, the human flow system is experimentally simulated to prove that the system is practical and can meet the general situations. The improved algorithm proposed in this study is applied to the statistics of the flow of people in public areas, which is of great

significance for the reasonable control of the flow of people and the maximization of the function of smart cities. The structure of this paper is as follows. The second section summarizes the current research background and the existing human flow statistics technology, and identifies the problems expected to be solved in this research. In Section 3, the flow statistics scheme based on machine vision and deep learning is implemented, and an improved algorithm introducing Smooth_L1 loss function is proposed based on YOLOv3 algorithm. In Section 4, the proposed improved algorithm is experimentally verified, which proves the advantages of the improved algorithm in pedestrian detection and recognition. The innovation and contribution of this work are as follows:

- In the module of human object detection, YOLOv3 detection algorithm is applied. On this basis, Smooth_L1 loss function is innovatively introduced to optimize the algorithm to avoid missing detection of objects.
- Considering that the short time tracking of the target has certain practical significance for the statistics of human flow, the Kalman filter and Mean Shift are combined to complete the iterative tracking first, and then the Kalman prediction is updated at last.
- The human flow statistics system based on machine vision and deep learning can effectively detect and track pedestrian targets, and provide help for the standardized development of commercial activities and leisure and entertainment projects in smart cities.

2 Research Background

“Smart City" emerged at the end of the last century to promote the sustainable development of cities. The new generation of information technology such as the Internet of Things and cloud computing has promoted the construction of “smart cities”, and to realize the sustainable development of the city has become one of the characteristics of the current era. The essence of a smart city is to realize information sharing, rational allocation of resources, and collaborative operations among different systems, based on which the decisions being conducive to urban development and management can be made to predict and respond to emergencies effectively. The emergence of urban complexes has played an important role in urban development and regional positioning. However, the ecological complexity has become increasingly significant as the central business districts gradually become saturated. The urban complex combines many functions such as residence, office, commerce, and entertainment, which influence and promote each other [13]. The urban functional morphology is essentially the interaction between the hierarchical units, and the urban complex should be integrated into it as an important functional node of the city, coexisting with the urban system. Thanks to the diverse functions of the urban complex, it can serve the people more effectively and increase people flow [14]. The digital technology makes crowd control possible in the rapid urbanization to mitigate the increasing congestion of human and road traffic and to jointly built sustainable, energy-saving, and livable cities (Figure 1).

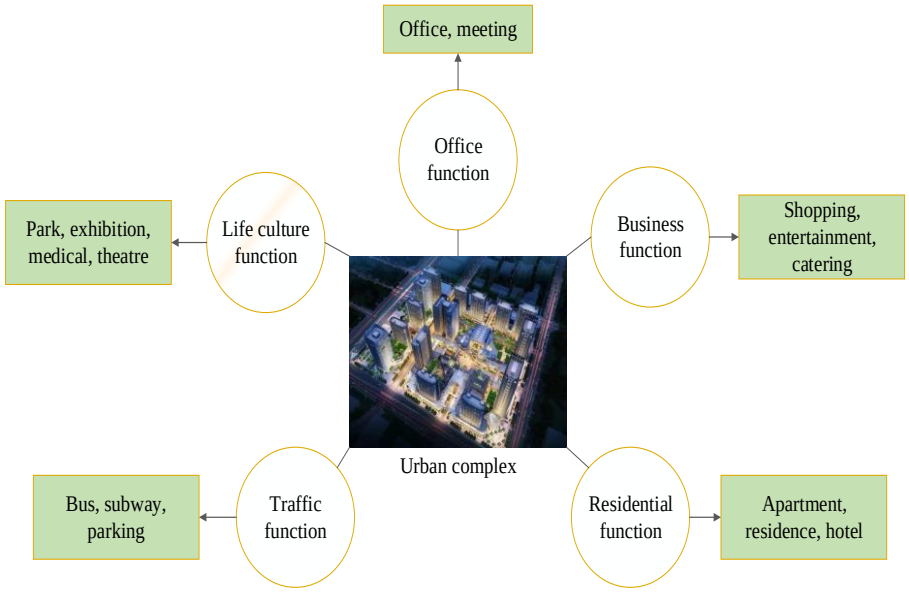


Fig. 1. Relationships between urban complexes in smart cities and their functions

While the living standards of people are steadily improving, the spiritual needs become higher and higher. The commerce and tourism industries in smart cities have achieved unprecedented development under the advancement of modern information technology and Internet technology. People have more entertainment projects, with more attention paid to the diversified development of life. Although the development of smart cities enriches the leisure and entertainment life of the masses, the tourists gathering brings a huge test to the crowd control of public areas such as scenic spots and shopping malls. The rapid social development and excessive people flow have caused many social problems and security risks. Consequently, real-time statistics of the number of people in the area are very important. Scientific control in the number of visitors can improve the management and service levels, prevent crowding, detention, and trampling, guaranteeing the personal safety and property safety. The people flow statistics is an important research method, and its related technologies have become common in some developed countries such as Europe and the United States. The people flow statistics enables the public facilities to operate within the maximum carrying capacity, and the people flow density is always controlled within a safe range.

Chen et al. (2022) [15] proposed the distributed probability adjusted confidence (DPAC) function, which can optimize the reliability of model prediction according to the actual situation. Based on the target confidence of the YOLOv4 network, the DPAC function was added to reduce or increase the confidence according to the target distribution and then output the final confidence. The proposed method showed obvious advantages in image-based traffic statistics. Zhang et al. (2020) [16] proposed a bus passenger flow statistics algorithm based on Single Shot MultiBox Detector (SSD) and Kalman filter to obtain passenger flow statistics from surveillance cameras on buses. In this method, the SSD model was modified into two types of models. Firstly, the two types of SSD models were trained by using the bus dataset, and then the passenger

position in each frame was detected by the model and tracked by the Kalman filter. Finally, according to the passenger trajectory, the traffic statistics of passengers getting on and off the bus were generated. Xie et al. (2020) [17] believed that from the perspective of spatial prediction measure, urban flow prediction can be divided into three categories, namely, city level, regional level, and road level. From the perspective of time prediction, urban flow prediction can be divided into three categories, such as short, medium, and long-term flow prediction. Many new methods have been proposed in recent years. The main methods can be divided into five categories: statistical methods, traditional machine learning methods, deep learning-based methods, reinforcement learning methods, and transfer learning methods.

It is concluded that most of the existing pedestrian detection technologies use machine learning technology to integrate pedestrian features and detect moving pedestrian objects in video. These methods generally find the basic features of moving objects through many samples, use neural networks and deep learning algorithms to train and classify various pedestrian features to form reliable data sets and models, and then bring these models and data sets into common detection scenarios. However, the information of moving objects is lost or mutated due to the dynamic changes of the background environment and the irregularity of pedestrian movement, which seriously affects the detection and tracking of moving pedestrians. Based on this, this work is developed by focusing on this difficulty.

3 The People Flow Statistics Based on Machine Vision and Deep Learning

3.1 Human Body Recognition

Innovation and digitalization have developed into new growth poles in recent years gradually. Countries around the world are actively looking for solutions for a balance among economy, technology, and society development. Generally, the intelligent video surveillance has become an inevitable trend in the development of the smart cities. It can extract and screen abnormal behaviors in video and issue early warnings in time, completely changing the passiveness that traditional surveillance can only "monitor" but not "control" [18,19]. From a technical point of view, there are mainly two categories of intelligent analysis technologies: one is to extract and detect the video targets with the image segmentation and foreground extraction technologies, and the other is to build a model of specific objects in the image using the pattern recognition technology. After a large amount of sample training, the detection of specific targets in the video is completed. The ultimate task of people flow statistics based on intelligent video surveillance is to count the number of people. Dynamic human dynamic characteristics refer to the distinctive characteristics that people show under dynamic conditions, such as the swing amplitude of shoulder, arm, elbow, leg, and foot of each person is different when walking. The difficulty lies in the capture of dynamic human target through video. Hence, target detection and human tracking should be completed.

Background modeling is the most common in target detection. It is a method to segment all moving targets in a scene under a fixed camera. The key is how to build a

background model from a video sequence. The classic background modeling methods include single Gaussian, inter-frame difference, mixed Gaussian, VIBE, and CodeBook. In practice, a suitable method is chosen by jointly considering the application scenario, the type of detection target, and the performance of the hardware platform [20]. The modeling principle is as follows. A background model is established based on the relevant parameters of the background, the difference operation of the current frame of the video is performed then, and the binarization result of the difference image is obtained. The foreground and background images can be distinguished by the gray value change degree, and the background image is the one with less gray value change. The specific process of the background modeling method is shown in Figure 2. In actual applications, the moving targets are detected through background modeling and then sent to the deep learning network for classification and recognition, which can ensure the real-time performance and the accuracy of classification [21].

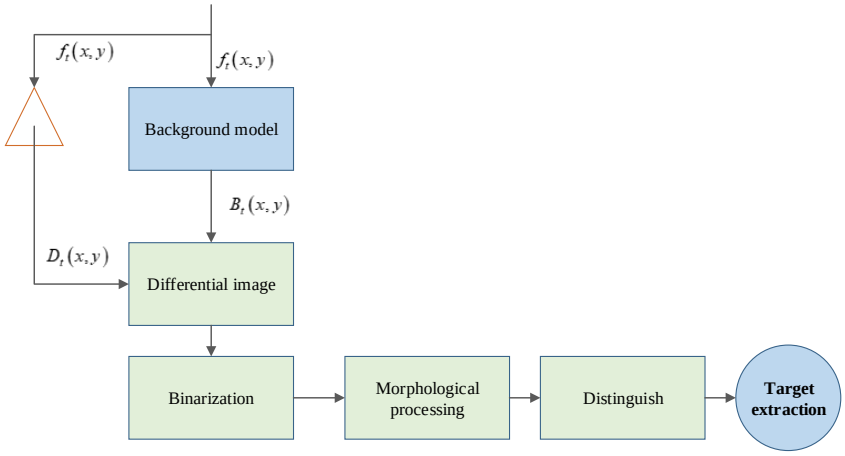


Fig. 2. The specific process of the background modeling method

Assuming that the current frame image is $f_t(x,y)$ and the background image is $B_t(x,y)$. (x,y) refers to the coordinate of image, and n represents the frame of the image. The image is input and compared with the background model to obtain the change information $D_t(x,y)$ of each pixel in the image to better detect the moving target. The accuracy of $B_t(x,y)$ directly affects the detection effects of moving targets. The premise of the background modeling method is to obtain the background image quickly and accurately. Further, $B_t(x,y)$ will change due to factors such as illumination, and the background model is updated in real time to adapt to the changing background.

The average background model is a background modeling algorithm that is more sensitive to environmental lighting changes and background multi-modality. The basic idea is as follows. If the coordinate location of the image is set as *(x,y)*, the average value of each pixel is calculated as its background modeling. When the current frame is detected, it only needs to subtract the average pixel *u(x,y)* of the same position in the background model from the pixel of the current frame *I(x,y)* to get the difference *d(x,y)*,

which will be compared with a threshold *TH*. It is deemed as foreground when the value is greater than the threshold, otherwise it is the background. Input image is a binary image.

The calculation process of the average background model is as follows [22].

$$d(x,y) = I(x,y) - u(x,y) \tag{1}$$

$$output(x,y) = \begin{cases} 1, & |d(x,y)| > TH \\ 0, & otherwise \end{cases} \tag{2}$$

The threshold TH can be determined by an adaptive algorithm, and the average value u_{diff} and standard deviation $diff_{std}$ of the inter-frame difference of each pixel are calculated. $I_t(x,y)$ represents the pixel value in the image (x,y) at time t, inter represents the interval between two frames, and $F_t(x,y)$ is as follows.

$$F_t(x,y) = \left| I_t(x,y) - I_{t-\text{inter}}(x,y) \right| \tag{3}$$

$$u_{diff}(x,y) = \frac{1}{M} \sum_{t=\text{inter}}^{M} F_t(x,y) \tag{4}$$

$$diff_{std}(x,y) = \sqrt{\frac{1}{M} \sum_{t-\text{inter}+1}^{M} \sum \left(F_t(x,y) + u_{diff}(x,y) \right)^2} \tag{5}$$

In the above equations, M refers to the number of iterations. Usually, M should be large enough to ensure the accuracy of u_{diff} and $diff_{std}$, and TH can be determined according to u_{diff} and $diff_{std}$.

$$\text{TH} = u_{diff}(x,y) + \beta \times diff_{std}(x,y) \tag{6}$$

where, the value of β is generally 2.

After the background modeling, the human body is detected from the video sequence and obtain the position information using the human body recognition. The human body recognition generally includes extracting the feature information of the target object and constructing the corresponding classifier to complete the target classification. The key of human body recognition method is to extract good features for clustering and design efficient classifier for behavior recognition. As a common feature in behavior recognition, the global feature is usually defined by background subtraction and tracking location, and then the interesting region is divided into a whole for description. Human body feature descriptors can be classified into three categories: low-level features, learning-based features, and mixed features. The global feature extraction regards the object as a whole. The target tracking algorithm locates the human body in the video, and then encodes the positioned target to form its global feature. However, the global feature extraction relies too much on the underlying visual processing, and the global feature is very sensitive to noise and occlusion in the image [23]. The local feature extraction is to extract the spatiotemporal interest points first in the video, and extract corresponding image blocks around these points. Finally, all image blocks are combined to jointly describe a specific action. The premise of local feature extraction is to have a sufficient number of stable interest points related to the action category. As a

result, the preprocessing process is more cumbersome [24]. The global feature extraction and local feature extraction are shown in Figure 3 and Figure 4, respectively. Global feature refers to the global feature information of each image, and the local feature is that in a certain area in the image. Finally, multiple local features are merged as the final feature.

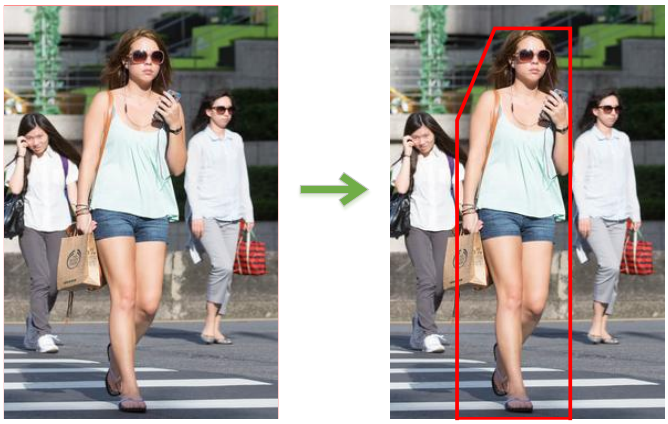


Fig. 3. Global feature extraction

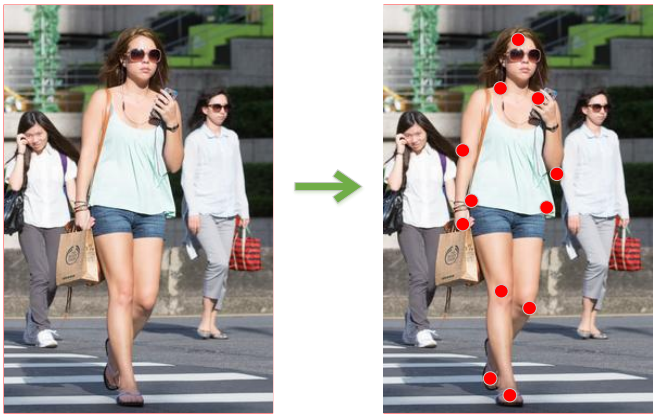


Fig. 4. Local feature extraction

To extract target features is the prerequisite for human body recognition, and the classifier is the key to classifying the image using the extracted target features [25]. After the HOG feature is extracted as the feature basis of human body, how to make the next decision is what we need to study. This section will make a detailed analysis of the training method of pedestrian classifier. After comparing different classifiers, we decided to use the support vector machine (SVM) with the most reliable sample partitioning effect as the classifier [26-28]. Many samples should be trained before classification. If the pedestrian classifier in Opencv library is directly used, it can be used. Due to the great difference between the training samples and the hypothetical scenarios in the system in this work, the recognition accuracy is not high. Therefore, it

is necessary to train the classifier with samples more in line with the actual scene to improve the accuracy to the required level. The major objective of this work is to count the pedestrian flow, so it is necessary to select a database containing various pedestrian poses for training.

3.2 Human Target Detection Algorithm Based on Yolov3 Network

Generally, the deep learning-based target tracking includes target tracking based on deep features and target tracking based on the deep neural network (DNN). The DNN-based target detection shows a wider application range and better effects. Convolutional neural network (CNN) is a multi-layer neural network. Once the distribution of training data and test data is different, the generalization ability of the network is also greatly reduced [29,30]. The YOLOv3 network (Figure 5) is a CNN for detection tasks. The v3 algorithm is formed on the basis of v1 and v2. The speed and accuracy can be controlled by changing the size of the model structure. YOLO only uses convolutional layers. It contains 75 convolutional layers, with skipping connection and up-sampling layers. Besides, it doesn't use any form of pooling, but uses a convolutional layer with a stride of 2 to down-sample the feature map, which is conducive to preventing loss of low-level features due to pooling. Layers 0 to 74 of YOLOv3 are the main network structures for feature extraction, and their convolutional layers are obtained by integrating various mainstream network structures with better convolutional layers. The 75th to 106th layers are feature interaction layers to predict boxes of different scales [31]. The priori detection system of YOLOv3 reuses the classifier or locator to perform the detection task, and applies the model to multiple positions and scales of the image, and those areas with higher scores can be regarded as the detection results.

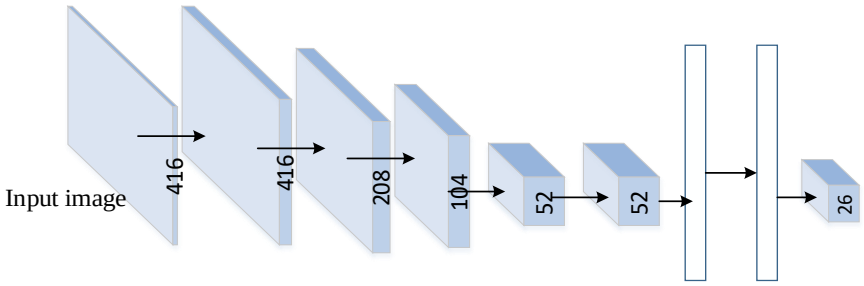


Fig. 5. YOLOv3 network structure

YOLOv3 is a detection network based on regression, and it judges whether there is an interest target in the image to extract target features through the network. YOLOv3 uses the previous 52 layers of darknet-53, with a large number of residual skip-floor connections applied. Meanwhile, the conv's stride is adopted for downsampling to reduce the gradient negative effects caused by pooling [32]. In this network structure, a convolution with a step size of 2 is used for downsampling. The image is divided into $S \times S$ by YOLOv3, each grid predicts bounding boxes at three different scales, and each bounding box predicts 4 coordinate values (t_x, t_y, t_w, t_h) and c classification predictions,

where (t_x, t_y) is the coordinate of the predicted bounding box relative to the grid unit, and (t_w, t_h) is the logarithmic ratio of the predicted bounding box size to the preset a priori box size. The bounding box is shown in Figure 6. The distance from the target grid to the upper left corner of the image is expressed as (c_x, c_y), and the position information of the bounding box is then obtained as follows [33].

$$b_x = \sigma(t_x) + c_x \tag{7}$$

$$b_y = \sigma(t_y) + c_y \tag{8}$$

$$b_w = p_w e^{t_w} \tag{9}$$

$$b_h = p_h e^{t_h} \tag{10}$$

$$confidence = \sigma(t_o) = p_r(object) \times IOU(b, object) \tag{11}$$

where, t_o is the target prediction, and $confidence$ is the confidence level to judge whether the bounding box is true, $p_r(object)$ judges whether there is a target in the grid, and $IOU(b, object)$ is the intersection ratio of the predicted bounding box and the true bounding box, defined as below (Figure 7).

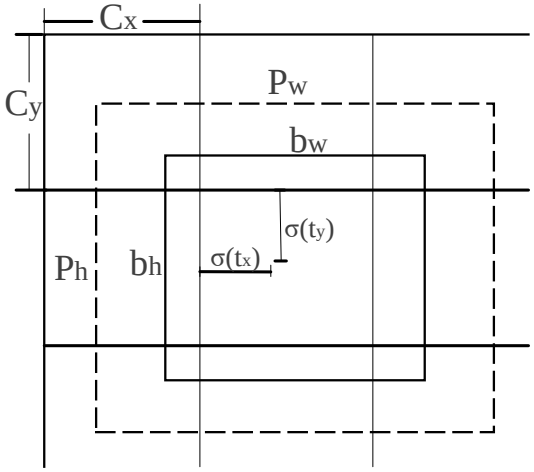
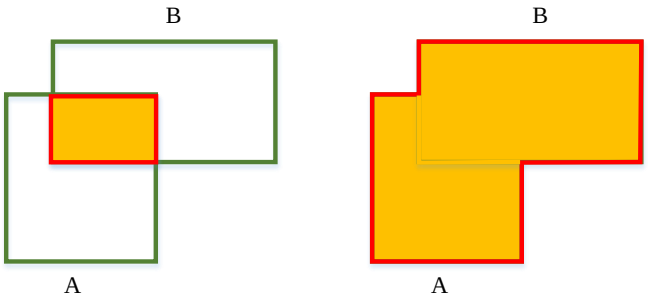


Fig. 6. Boundary box

The Smooth_L1 loss function is used in the Faster R-CNN algorithm. Compared with the L2 loss function, Smooth_L1 is not sensitive to abnormal values. The derivative of the average absolute value error of L1 loss function is constant, so when the loss value is small, the gradient obtained is relatively large, which may cause model oscillation and is not conducive to convergence. The L2 loss function, due to the square operation, will amplify the error when the difference between the predicted value and the true value is greater than 1. Especially when the input value of the function is far

from the center value, the gradient is large when the gradient descent method is used to solve it, which may cause gradient explosion. At the same time, when there are multiple outliers, these points may occupy the main part of Loss, and many effective samples need to be sacrificed to compensate for it. Therefore, in Bounding box regression of object detection, Smooth L1 Loss is considered early for the following reasons: (1) compared with L1 Loss, it can converge faster; and (2) compared with L2 Loss, it is not sensitive to outliers and outliers, and the gradient change is relatively smaller, so it is not easy to run and fly during training.



$$IOU = \frac{A \cap B}{A \cup B}$$

Fig. 7. Schematic diagram of IOU definition

In this work, the Smooth_L1 loss function is introduced to calculate the loss of the width and height of the bounding box to overcome the model instability caused by large parameter changes with abnormal values. The Smooth_L1 loss function essentially combines the L1 and the L2 loss functions. L2 is dominate when the difference between the predicted value and the target value is small; while L1 is active when the predicted value differs greatly from the target value. The curve graphs of the three loss functions are shown in Figure 8.

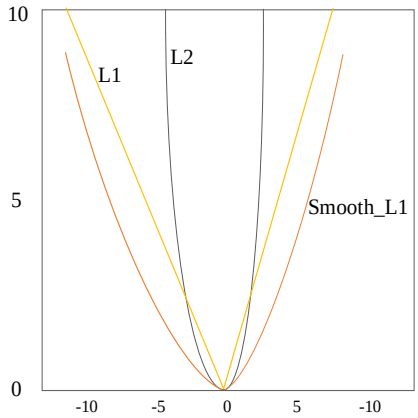


Fig. 8. Three loss functions of L1, L2, and Smooth_L1

The Smooth_L1 loss function can be expressed as follows:

$$Smooth_{L1}\left(b_i - f\left(a_i\right)\right) = \begin{cases} 0.5\left(b_i - f\left(a_i\right)\right)^2, \left(\left|b_i - f\left(a_i\right)\right| < 1\right) \\ \left|b_i - f\left(a_i\right)\right| - 0.5, \left(\left|b_i - f\left(a_i\right)\right| \geq 1\right) \end{cases} \quad (12)$$

The detection algorithm based on deep learning gives higher requirements on the data set, and the trained network weight model shows a higher accuracy rate. The characterization model is selected based on the characteristics of the whole human body, and detection results depend on the completeness of the human target in the video. However, since human targets are easily occluded, good detection results are usually not achieved. In addition, there are characterization models for head information, facial features, etc., which can reduce the occlusion of human targets, whereas missed detection will occur due to the limited camera capture. In this work, the representative pedestrian detection sample set ImageNet is used to perform high-resolution pre-training on YOLOv3 to obtain the classification network. The VOC data set is then used for training to obtain the final training model. All images in the ImageNet image data set have at least one frame. The detection for 200 targets has produced 470,000 images, and each image has 1.1 targets on average. The VOC data set mainly includes four categories, namely people, common animals, transportation vehicles, and indoor furniture. It mainly serves three types of tasks: image classification, object detection and recognition, and image segmentation. The performance of the pedestrian detection algorithm is evaluated using the P-R curve.

3.3 Multi-Head Target Tracking Based on Kalman Filter

After the human targets are extracted and detected based on deep learning algorithms, the pedestrians are tracked in the video for a certain period to count the specific number of pedestrians entering and exiting in detail and to improve the accuracy. Usually, multiple pedestrians appear in the same scene at the same time. Hence, a multi-target tracking algorithm is employed to track multiple pedestrian targets at the same time [34]. The Kalman filter algorithm is a mainstream algorithm with an excellent processing effect on Gaussian process noise during target tracking. The Kalman filter includes a state model and an observation model. The state model is used to describe the relationship between continuous-time states, which can be expressed as equation (13), and the observation model can be expressed as equation (14).

$$x_k = F_k x_{k-1} + B_k u_k + W_k \quad (13)$$

$$z_k = H_k x_k + v_k \quad (14)$$

where, x_k and x_{k-1} represent the state vector at time k and k-1, respectively. F_k represents the state transition matrix, which is used to describe the relationship between the system state at time k and k-1. B_k is the control input matrix, which represents the connection between the control input parameters and the state vector. W_k is the process noise vector, which can be regarded as a normal distribution that obeys the zero mean value. z_k represents the observation vector, H_k represents the transition matrix, and v_k is the observation noise.

The Kalman filter algorithm has two basic hypotheses. I. A sufficiently accurate model is a linear (or time-varying) dynamic system excited by white noise; and II. Each measurement signal contains additional white noise components [35]. When the above hypotheses are met, the Kalman filter algorithm can be applied. The Kalman filter mainly includes two stages: prediction and update. The prediction stage is to predict the next time information based on the current state information pair, and the update stage is to update the target position based on the predicted information. The update equation for the Kalman filter time can be expressed as follows [36].

$$\hat{x}_{\bar{k}} = A\hat{x}_{k-1} + Bu_{k-1} \tag{15}$$

$$P_{\bar{k}} = AP_{k-1}A^T + Q \tag{16}$$

$$K_k = \frac{P_{\bar{k}}H^T}{HP_{\bar{k}}H^T + R} \tag{17}$$

$$\hat{x}_k = \hat{x}_k + K_k\left(z_k - H\hat{x}_{\bar{k}}\right) \tag{18}$$

$$P_k = \left(I - K_k H\right)P_{\bar{k}} \tag{19}$$

where, $\hat{x}_{k-1}$ and $\hat{x}_k$ represent the posterior state estimates at t-1 and t, respectively, and are the filtering results; $\hat{x}_{\bar{k}}$ represents the prior state estimates at t; P_{k-1} and P_k represent the posterior estimate covariance at t-1 and t, respectively; $P_{\bar{k}}$ is the a priori estimated covariance at time t. H is the conversion matrix from state variable to observation, which connects the state and observation; z_k is the observation value, and is the input of filtering; K_k is the filter gain matrix; A is the state transition matrix; Q is the process excitation noise covariance; R is the measurement noise covariance; B is the matrix that converts the input to state.

The Kalman filter tracking algorithm only needs to input the current state information as the initial data to predict the target motion state at the next moment, and the previous moment data can be discarded after each prediction update [37-39]. Therefore, the Kalman filter has a good tracking effects on continuous moving targets. However, when there are multiple targets in the video and there is interference between each other, it will cause chaos in multi-target tracking or target loss. The Mean Shift algorithm relies on the probability density value of the current human head target feature sample in the region, so it can be used for clustering, image segmentation, and target tracking. Therefore, the Mean Shift algorithm is incorporated into the Kalman filter to tract the target first. When the target is lost, the current tracking information of the Kalman filter is brought into the Mean Shift algorithm for iterative tracking [40-42]. Finally, the Kalman prediction update is performed. Since the offset of the targets in two adjacent frames is very small and the target box B1 of the current frame F1 is known, a heuristic approach is to still select the known B1 area in the next frame F2 to infer the offset required by the target box according to the similarity between the B1 area in F2 and the B1 area in F1, and then get the target box B2 of F2. This is what the MeanShift algorithm needs to solve. Combined with MATLAB source code, the

tracking process of MeanShift is shown in Table 1. The flow of the optimized pedestrian multi-target tracking algorithm is shown in Figure 9.

Table 1. The tracking process of the Mean Shift algorithm

```
To calculate the histogram of the candidate area
hist2 = zeros(1,4096);
To calculate the histogram of the candidate area
for i = 1:a
   for j = 1:b
      % rgb color space quantization 16 is *16*16 bins
      q_r = fix(double(current_temp(i,j,1))/16); % fix integral function is taken as it approaches 0
      q_g = fix(double(current_temp(i,j,2))/16);
      q_b = fix(double(current_temp(i,j,3))/16);
      q_temp(i, j) = q_r*256 + q_g*16 + q_b;    % to set the proportion of red, green, and blue
components to each pixel
      % to calculate the weight of each pixel in the histogram
      hist2(q_temp(i, j)+1) = hist2(q_temp(i, j)+1) + m_wei(i,j);
   end
end
hist2 = hist2 * C;  The above steps complete the histogram of the target kernel function
```

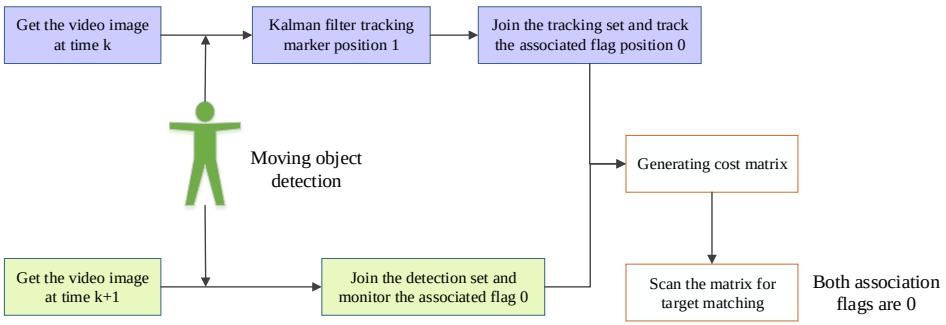


Fig. 9. Optimized pedestrian multi-target tracking algorithm

3.4 Experimental Design and Model Performance Evaluation

The machine learning is combined with the Kalman filter algorithm in this work to complete the detection and tracking of human head targets for people flow statistics. The pedestrian motion video sequence in the surveillance video is acquired first, followed by the segmentation of the area where the pedestrian is located. The YOLOv3 network is used to detect the extracted target to identify all the head targets. Finally, the optimized pedestrian multi-target tracking algorithm is used to achieve the people flow statistics. The software development environment of the people flow statistics system is Visual Studio 2013, together with the OpenCV image processing library. Databases for image and video processing for the OpenCV are the most comprehensive, which can satisfy the video extraction, image morphology processing, and target detection. The face recognition in the video includes two steps. Step 1: the image size is adjusted to

92×112, that is, the same size as the trained image. Step 2: the predict function is used to detect the face in the image, and the function concerns two elements: the label to identify the individual and the confidence. A smaller confidence indicates a higher degree of match. 0 means a perfect match, but the score mechanism of the confidence is different for of different algorithms.

The hardware platform of the traffic statistics system can be divided into five parts: front-end acquisition module, signal transmission module, information storage module, computer processing module, and output display module. The front-end acquisition module is the premise and foundation of the whole system, and the scene in the front-end detection area is converted into image signals by CCD cameras. The signal transmission transmits the collected image signal to the processing equipment by Ethernet. The information storage module mainly refers to the server, which is used to store the image data collected by the front end for later search; the computer processing module mainly uses workstations or high-performance computers, which is the core part of the whole system and is used to analyze and process image signals. The output module can display front-end monitoring images and real-time traffic data.

To detect the accuracy, the original video is produced in the campus library and the teaching building, and the video resolution is 640×480. The tracked pedestrians are marked with the yellow line frame to better identify pedestrian targets and facilitate the pedestrian order distinction, based on which the detection effects are then evaluated.

4 Results and discussion

4.1 YOLOv3 +Smooth_L1 model

In this work, the weight model of the YOLOv3 algorithm is applied to detect the pedestrians in real life scenes. The model is trained using the VOC data set, which uses the characteristics of the whole human body as the representation information. Adam is used as an optimizer to minimize the loss function, $\beta_1 = 0.9$ $\beta_2 = 0.999$. Training can be carried out when batchsize is set to 2, so the experiments are carried out when batchsize=2. An epoch refers to a complete training of all training data. The initial learning rate is set to 0.0001. After 10 epochs, it is reduced to one-tenth of the current value. When 30 epochs are finished, the model has converged. The specific hyperparameter setting of the model is listed in Table 2 below.

Table 2. The specific hyperparameter setting of the model

Hyperparameters	Value
β_1	0.9
β_2	0.999
Batchsize	2
Initial learning rate	0.0001

Entrance of the campus teaching building is selected as the video scene for detection, as shown in Figure 10. It was evident that the detection effects of the YOLOv3 model are good, but pedestrians with serious occlusion are missing. When the people flow

density is intensive, the occlusion of human targets will be more serious, bringing great difficulty for people flow statistics.

Fig. 10. Pedestrian detection results based on the YOLOv3 model

To further reduce the missed detection of the YOLOv3 model, the Smooth_L1 loss function is introduced to calculate the width and height of the bounding box. The model established is then trained and tested, and compared with the weight model obtained using the L2 loss function. The P-R curve obtained is shown in Figure 11. It is evident that the P-R curve of the Smooth_L1 loss function is closer to the upper right, indicating that the model trained is superior in accuracy and recall rates and is more efficient.

The experimental results demonstrate that the mAP value of the improved YOLOv3 algorithm by introducing Smooth_L1 loss function is higher than the optimal mAP value of YOLOv3 algorithm. Mean average precision (mAP) refers to the average mean recognition accuracy of each class of objects when the model recognizes multiple classes of objects. The higher the mAP value, the higher the recognition accuracy of the model. The average precision (AP) is the area under the P-R curve of the pedestrian detection algorithm. The higher the AP value, the better the detection performance of the algorithm. The data calculation results of the training log are shown in Table 3. It can be observed that, compared with YOLOv3, the recall rate of pedestrian target detection of YOLOv3 algorithm improved by introducing Smooth_L1 loss function is increased from 82% to 84%. In the study of Zhang et al. (2021), a Caps-YOLO model based on YOLOv3 real-time object detection model was proposed for pedestrian problem detection. In this model, dense connections are used to replace the shortcut connections in the original network, and dense block components are constructed to improve the utilization of feature maps. This is consistent with the original intention of choosing the YOLO model in this work, and also confirms the reliability of the proposed model in pedestrian detection.

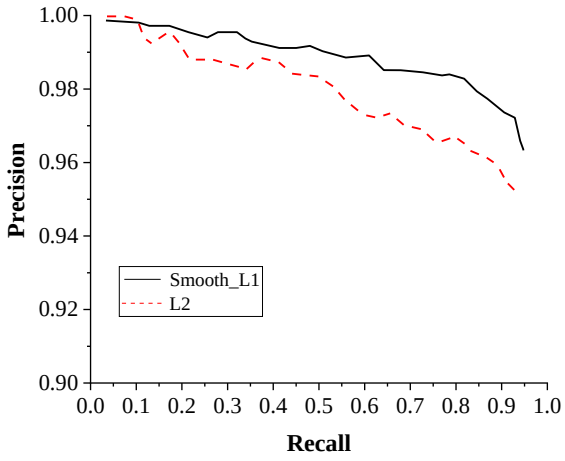


Fig. 11. PR curves of different loss function models

Table 3. Comparison of target detection results between YOLOv3 and YOLOv3+ Smooth_L1

Algorithm	AP/100%	AP50/%	AP60/%	P/100%	R/%
YOLOv3	23.2	72.9	46.7	53.5	82
YOLOv3+Smooth_L1	27.2	76.3	52.7	64.5	84

Furthermore, the proposed algorithm is compared with the human FLOW statistics method based on contour rules on the flow dataset. The FLOW dataset consists of 1148 frames and includes 9 persons. It simulates the scene of rapid vertical flow of people, such as underground passage and building entrance and exit, and includes part of lateral movement. In this work, the sequence is used to evaluate the statistical accuracy of the proposed algorithm in the case of rapid movement and direction change. The comparison results are shown in Table 4. It can be observed that the F1 score of the algorithm in this work is more than 95% and even as high as 100% in the vertical counting of rapid personnel flow, which is obviously better than the human flow detection method based on contour rules.

Table 4. Performance comparison of YOLOv3+Smooth_L1 and contour rule-based human flow statistics method

Algorithm	Direction	Precision (%)	Recall (%)	F1 (%)	Tracking time (s)
Contour rule-based human flow statistics method	Longitudinal (upward)	100	90	95	0.009
YOLOv3+Smooth_L1	Vertical (down)	94	88	88	0.009
	Horizontal (to the left)	100	100	100	0.014
	Landscape (to the right)	95	100	95	0.014

4.2 Pedestrian Tracking Results Based on Kalman Filter

In this work, the Mean shift algorithm is incorporated into the Kalman filter to track the moving pedestrian targets, which can correlate images of different frames of the same target to avoid no repeated count. At the same time, the movement direction of the target is identified to count the specific number of targets accurately. Figure 12 suggests that the system can track the two targets effectively, avoiding the target distortion and target loss.

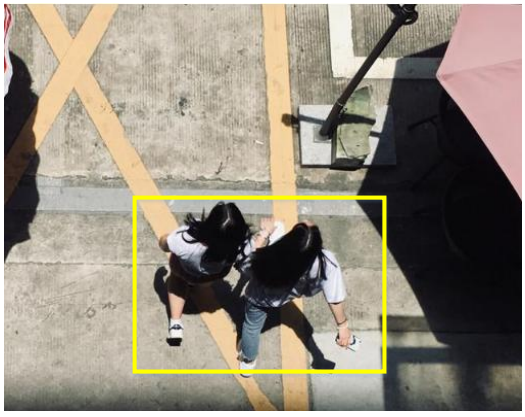


Fig. 12. Pedestrian tracking effects based on Kalman filter

In this work, people on the campus library and the teaching building are counted, and the results are shown in Figure 13. 7,000 original images are collected in the library, involving a total of 88 people. Of them, 82 people are detected, and the detection accuracy reaches 93.18%. There are 12,200 original images collected in the teaching building, involving a total of 149 people. Of them, 139 people are detected, and the detection accuracy is 93.29%. Obviously, tracking the moving target by undertaking the result of the Kalman filter as the input value of the Mean Shift algorithm can reduce the target loss caused by the occlusion effectively, ensuring high target tracking efficiency.

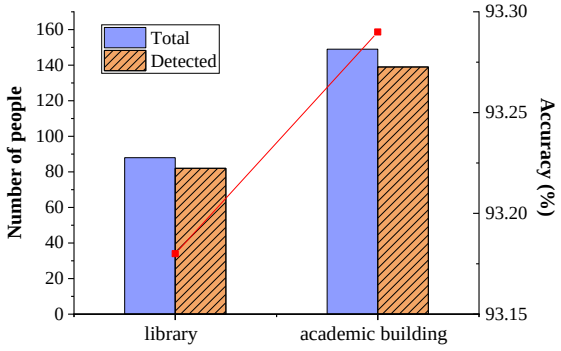


Fig.13. Pedestrian detection results in different scenarios

5 Conclusion

A smart city is characterized by the connection of the scattered and discontinuous parts of the city and multiple functions. With the continuous development of the city, the existing functions may not be able to adapt to the current public life and needs. The construction of smart city brings about crowd congestion and ineffective control. To alleviate the pressure brought by the large people flow, the technologies used for people flow statistics are analyzed, among which the human target recognition and target tracking are most important. In the human recognition module, the human body is first detected from the video sequence based on machine vision, and the position information is then obtained, and the extracted global features and local features are merged as the final feature of the target. The short-term tracking of the target has a certain practical significance for people flow statistics, so the Mean Shift algorithm is incorporated into the Kalman filter to perform iterative tracking, and then the Kalman prediction is performed.

The innovation of this work lies in the application of YOLOv3 detection algorithm in the human object detection module, and on this basis, Smooth_L1 loss function is introduced to optimize the algorithm to avoid missing detection of the object. The experimental results reveal that the YOLOv3 model shows a good target detection effect, but pedestrians with serious occlusion are missing. After the Smooth_L1 loss function is introduced, the P-R curve is closer to the upper right side, which can avoid the model instability caused by large parameter changes with abnormal values. By incorporating the Mean shift algorithm into the Kalman filter to track moving pedestrian targets, high target tracking efficiency is achieved in both the campus library and the teaching building. This work contributes that the human flow statistics system based on machine vision and deep learning can effectively detect and track pedestrian targets and provide help for the standardized development of commercial activities and leisure and entertainment projects in smart cities. There are still some shortcomings in this work. The computation amount of the system algorithm is huge, and the processing ability of the computer is high. In addition, the system shows poor response ability to fast pedestrians, which is easy to miss, resulting in reduced accuracy. Therefore, the algorithm will continue to be optimized in the subsequent research to reduce the computational complexity of the algorithm.

Acknowledgment. This work was supported by the Natural Science Research Project of Anhui Province, China (KJ2021A1121).

References

1. Meijer, A., Bolívar, M. P. R.: Governing the smart city: a review of the literature on smart urban governance. International review of administrative sciences, Vol. 82, No. 2, 392-408. (2016)
2. Meijer, A., Thaens, M.: Urban technological innovation: Developing and testing a sociotechnical framework for studying smart city projects. Urban Affairs Review, Vol. 54, No. 2, 363-387. (2018)

3. Romão, J., Kourtit, K., Neuts, B., Nijkamp, P.: The smart city as a common place for tourists and residents: A structural analysis of the determinants of urban attractiveness. Cities, Vol. 78, 67-75. (2018)
4. Guevara, S., Singh, Y., Shores, A., Mercado, J., Postigo, M., Garcia, J., Newell, B.: Development of a Pilot Smart Irrigation System for Peruvian Highlands. Journal of Contemporary Water Research & Education, No. 171, 49-62. (2020)
5. El-Sayed, H., Chaqfa, M., Zeadally, S., Puthal, D.: A traffic-aware approach for enabling unmanned aerial vehicles (UAVs) in smart city scenarios. IEEE Access, Vol. 7, 86297-86305. (2019)
6. Rehman, T. U., Mahmud, M. S., Chang, Y. K., Jin, J., Shin, J.: Current and future applications of statistical machine learning algorithms for agricultural machine vision systems. Computers and electronics in agriculture, Vol. 156, 585-605. (2019)
7. Williams, H. A., Jones, M. H., Nejati, M., Seabright, M. J., Bell, J., Penhall, N. D., et al.: Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. biosystems engineering, Vol. 181, 140-156. (2019)
8. Ouyang, W., Zhou, H., Li, H., Li, Q., Yan, J., Wang, X.: Jointly learning deep features, deformable parts, occlusion and classification for pedestrian detection. IEEE transactions on pattern analysis and machine intelligence, Vol. 40, No. 8, 1874-1887. (2017)
9. Arnold, E., Al-Jarrah, O. Y., Dianati, M., Fallah, S., Oxtoby, D., Mouzakitis, A.: A survey on 3d object detection methods for autonomous driving applications. IEEE Transactions on Intelligent Transportation Systems, Vol. 20, No. 10, 3782-3795. (2019)
10. Mateus, A., Ribeiro, D., Miraldo, P., Nascimento, J. C.: Efficient and robust pedestrian detection using deep learning for human-aware navigation. Robotics and Autonomous Systems, Vol. 113, 23-37. (2019)
11. Ullah, W., Ullah, A., Hussain, T., Muhammad, K., Heidari, A. A., Del Ser, J., et al.: Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data. Future Generation Computer Systems, Vol. 129, 286-297. (2022)
12. Dong, K., He, Z., Shen, D., Huang, Z., Chen, X.: Design of Pedestrian Flow Detection System for Playground Entrance/Exit. International Core Journal of Engineering, Vol. 7, No. 9, 332-336. (2021)
13. Capra, C. F.: The Smart City and its citizens: Governance and citizen participation in Amsterdam Smart City. International Journal of E-Planning Research (IJEPR), Vol. 5, No. 1, 20-38. (2016)
14. Muvuna, J., Boutaleb, T., Baker, K. J., & Mickovski, S. B.: A methodology to model integrated smart city system from the information perspective. Smart Cities, Vol. 2, No. 4, 496-511. (2019)
15. Chen, W., Wu, G., Jung, H.: An Optimization Method for Personnel Statistics Based on YOLOv4+ DPAC. Applied Sciences, Vol. 12, No. 17, 8627. (2022)
16. Zhang, Y., Tu, W., Chen, K., Wu, C. H., Li, L., Ip, W. H., Chan, C. Y.: Bus passenger flow statistics algorithm based on deep learning. Multimedia Tools and Applications, Vol. 79, No. 39, 28785-28806. (2020)
17. Xie, P., Li, T., Liu, J., Du, S., Yang, X., Zhang, J.: Urban flow prediction from spatiotemporal data using machine learning: A survey. Information Fusion, Vol. 59, 1-12. (2020)
18. Wu, H., Gao, C., Cui, Y., Wang, R.: Multipoint infrared laser-based detection and tracking for people counting. Neural Computing and Applications, Vol. 29, No. 5, 1405-1416. (2018)
19. Kang, D., Ma, Z., & Chan, A. B.: Beyond counting: Comparisons of density maps for crowd analysis tasks-counting, detection, and tracking. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 29, No. 5, 1408-1422. (2018)
20. Gao, C., Wang, L., Xiao, Y., Zhao, Q., Meng, D.: Infrared small-dim target detection based on Markov random field guided noise modeling. Pattern Recognition, 2018, 76, Vol. 76, 463-475. (2018)

21. Bai, X., Bi, Y.: Derivative entropy-based contrast measure for infrared small-target detection. IEEE Transactions on Geoscience and Remote Sensing, 2018, 56(4), Vol. 56, No. 4, 2452-2466. (2018)
22. Lu, Z. M., Zhu, F. C., Gao, X. Y., Chen, B. C., Gao, Z. G.: In-situ particle segmentation approach based on average background modeling and graph-cut for the monitoring of l-glutamic acid crystallization. Chemometrics and Intelligent Laboratory Systems, Vol. 178, 11-23. (2018)
23. Vishnu, V. C. M., Rajalakshmi, M., Nedunchezhian, R.: Intelligent traffic video surveillance and accident detection system with dynamic traffic signal control. Cluster Computing, Vol. 21, No. 1, 135-147. (2018)
24. Zhang, H. B., Zhang, Y. X., Zhong, B., Lei, Q., Yang, L., Du, J. X., Chen, D. S.: A comprehensive survey of vision-based human action recognition methods. Sensors, Vol. 19, No. 5, 1005. (2019)
25. Luo, F., Guo, W., Yu, Y., Chen, G.: A multi-label classification algorithm based on kernel extreme learning machine. Neurocomputing, Vol. 260, 313-320. (2017)
26. Raj, R. J. S., Shobana, S. J., Pustokhina, I. V., Pustokhin, D. A., Gupta, D., Shankar, K. J. I. A.: Optimal feature selection-based medical image classification using deep learning model in internet of medical things. IEEE Access, Vol. 8, 58006-58017. (2020)
27. Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., Benediktsson, J. A.: Deep learning for hyperspectral image classification: An overview. IEEE Transactions on Geoscience and Remote Sensing, Vol. 57, No. 9, 6690-6709. (2019)
28. Lv, N., Chen, C., Qiu, T., Sangaiah, A. K.: Deep learning and superpixel feature extraction based on contractive autoencoder for change detection in SAR images. IEEE transactions on industrial informatics, Vol. 14, No. 12, 5530-5538. (2018)
29. Xue, W., Jiang, T.: An adaptive algorithm for target recognition using Gaussian mixture models. Measurement, Vol. 124, 233-240. (2018)
30. Zhao, L., Li, S.: Object Detection Algorithm Based on Improved YOLOv3. Electronics, Vol. 9, No. 3, 537. (2020)
31. Pang, L., Liu, H., Chen, Y., Miao, J.: Real-time Concealed Object Detection from Passive Millimeter Wave Images Based on the YOLOv3 Algorithm. Sensors, Vol. 20, No. 6, 1678. (2020)
32. Xu, D., Wu, Y.: Improved YOLO-V3 with DenseNet for Multi-Scale Remote Sensing Target Detection. Sensors, Vol. 20, No. 15, 4276. (2020)
33. Zhao, L., Li, S.: Object detection algorithm based on improved YOLOv3. Electronics, Vol. 9, No. 3, 537. (2020)
34. Wu, Z., Fu, M., Xu, Y., Lu, R.: A distributed Kalman filtering algorithm with fast finite-time convergence for sensor networks. Automatica, Vol. 95, 63-72. (2018)
35. Yang, H., Wang, J., Miao, Y., Yang, Y., Zhao, Z., Wang, Z., et al.: Combining Spatio-Temporal Context and Kalman Filtering for Visual Tracking. Mathematics, Vol. 7, No. 11, 1059. (2019)
36. Jover J M, Kailath T.: A parallel architecture for Kalman filter measurement update and parameter estimation. Automatica, Vol. 22, No. 1, 43-57. (1986)
37. Zhang, Z., Fu, K., Sun, X., Ren, W.: Multiple target tracking based on multiple hypotheses tracking and modified ensemble Kalman filter in multi-sensor fusion. Sensors, Vol. 19, No. 14, 3118. (2019)
38. Huang, M., Guan, W., Fan, Z., Chen, Z., Li, J., Chen, B.: Improved target signal source tracking and extraction method based on outdoor visible light communication using a cam-shift algorithm and kalman filter. Sensors, Vol. 18, No. 12, 4173. (2018)
39. Fang, Y., Yu, L., Fei, S.: An Improved Moving Tracking Algorithm with Multiple Information Fusion Based on 3D Sensors. IEEE Access, Vol. 8, 142295-142302. (2020)
40. Xie, Z., Guan, W., Zheng, J., Zhang, X., Chen, S., Chen, B.: A high-precision, real-time, and robust indoor visible light positioning method based on mean shift algorithm and unscented Kalman filter. Sensors, Vol. 19, No. 5, 1094. (2019)

41. Huang, M., Guan, W., Fan, Z., Chen, Z., Li, J., Chen, B.: Improved target signal source tracking and extraction method based on outdoor visible light communication using a cam-shift algorithm and kalman filter. Sensors, Vol. 18, No. 12, 4173. (2018)
42. Zhang, C., Luo, K., Gu, S., Chen, L., Xia, Z., Gao, J.: Caps-YOLO: Pedestrian detection method of complex posture combined with capsules network. Journal of Flow Visualization and Image Processing, Vol. 28, No. 3. (2021)

Baofeng Yao was born in Bengbu，Anhui, P.R. China, in 1980. He received his B.S. degree in computer science from the Huaibei Normal University, Huaibei, China in 2001, M.S. degree from the Hefei University of Technology, Hefei, China in 2009. Now, he is an Associate Professor with the School of Computer Science and Information Engineering, Bengbu University. His research interest include machine learning, information security and big data analysis. E-mail: ybf@bbc.edu.cn

Shijun Liu was born in 1980 in Wuhe, Bengbu, Anhui, China. He received his B.S.degree in computer science from the Anqing Normal University,Anqing，China in 2004， M.S.degree from the Southeast University，Nanjing, China in 2016.Now， he is a Lecturer with the School of Computer Science and Information Engineering, Bengbu University. His research interest include machine learning, computer vision, privacy protection algorithms and deep learning. E-mail: liushijun@bbc.edu.cn

Lei Wang was born in Suzhou，Anhui, P.R. China, in 1978. He received his B.S. degree in computer science from the Huaibei Normal University, Huaibei, China in 2001, M.S. degree from the Hefei University of Technology, Hefei, China in 2009. Now, he is an Lecturer with the School of Computer Science and Information Engineering, Bengbu University. His research interest include machine learning, information security and big database analysis. E-mail: bbxywl@bbc.edu.cn

Received: August 13, 2022; Accepted: December 25, 2022.

Computer Science and Information Systems 20(2):701–722 https://doi.org/10.2298/CSIS220703019Z

Using Neural Network to Automatic Manufacture Product Label in Enterprise under Iot Environments

Kai Zhang[1] and Chongjie Dong[2]

[1] School of information, Guangdong Communication Polytechnic,
Guangzhou, 510650, China
gdcpzk@qq.com
[2] Department of Computer Engineering, Dongguan Polytechnic,
Dongguan, 523808, China
dchj2020@foxmail.com

Abstract. When the manufacturing industry is dealing with information technology, it has to face a large number of parameters and frequent adjustments. This study proposed artificial intelligence models to find out the hidden rules behind a large number of customized labels, through data processing and model building. Model and parameter experiments are used to improve the effectiveness of artificial intelligence models, and the method of cyclic testing is adopted to increase the diversity of the test set. The results of this paper, we integrate each stage and an auxiliary decision-making is established. The contributions of this paper, can improve the problem with reducing production line shutdown and improve factory productivity. The accuracy rate of the artificial intelligence model can be increased to 95%. The number of stoppages is reduced from 4 times to 1 time per month. Under full capacity, this assist the decision-making system can reduce loss cost.

Keywords: Artificial intelligence (AI); Machine learning; Random forest; Neural network; Automatic product label.

1. Introduction

With the development of science and technology, in order to save manpower, reduce costs, increase production capacity, and improve quality, companies have successively introduced new technologies and new concepts, from the early information and automation to today’s smart factories, Industry 4.0, artificial intelligence as the goal of intelligence [25]. To implement intelligence, automation must be carried out first. Based on automation, intelligence can make decisions based on the actual conditions of the site, and deploy the automation work sequence to optimize efficiency and productivity. To implement automation, information must be carried out first, based on information only by automation can judge the correct position, use the best path, make appropriate actions, replace labor and improve efficiency. Only in this way can maximum value of existing resources, difficulties can be reduced and benefits can be improved. However, when the manufacturing industry deals with information, it is not only necessary to import all the parameters into the system. The factory has a variety of products,

machines, manufacturing processes, post-station data, and the production line is divided into production and assembly, testing, packaging, some parameters have a reciprocal and complex relationship with each other. In addition, it is necessary to deal with the problem of frequent adjustment and change of parameters [31, 25].

It is possible to design card control check logic for products with higher severity, more value and longer cycle are including data setting limit card control and post-setting check comparison. In the case of frequent changes to specifications and revisions, the information department is time-consuming and laborious, and cannot meet the overall needs. In response to this problem, this research attempts to use artificial intelligence models to establish an auxiliary decision-making system. When the user sets an error, a warning will be displayed to reduce the error rate and improve the problem [15]. In the face of a difficult to control environment with diversified customers, diversified products, variable specifications, and short production cycles in the manufacturing industry, the only thing that remains constant is always changing. Under such circumstances, the setting of the production line system is very complicated. Under the premise of sharing the settings as much as possible, there are still many customized special rules. It is easy for the setting personnel to confuse, and the setting is wrong due to accidental setting. Once the setting is wrong, it will cause loss. Defective products require maintenance, heavy industry, or even scrapping. If there is an error, the product will be scrapped in the slightest, and the line will be stopped in the serious, resulting decrease factory productivity, and even delayed shipments. In the case of full capacity, the line is stopped due to the customized label problem, which takes about half an hour to solve each time. The loss is nearly every year ten million dollars. Decrease in factory productivity will affect quality and delivery, which will cause customers to reduce orders and revenue will lead to unhappy shareholders and the board of directors; unhappy and negative attitudes at the bottom of the company will lead to a decline in factory productivity [43, 24].

This study uses artificial intelligence models to try to predict the setting value in the changing environment to reduce the setting error rate and maintain the stable productivity of the factory [1, 5]. On the technical side: When facing a single customer and a single product, if the specifications can be clarified, the final answer should be limited to a few possibilities based on conditions. When the answer falls outside these possible ranges, the probability of error is extremely high. However, it is difficult to be familiar with all the rules due to human experience, and it is difficult to pass on the experience of senior staff to new recruits through education and training. However, the information department of general enterprises has limited resources, and it is difficult to develop all specifications with limited time and manpower, not to mention clarification of specifications, meetings, and communication time cost. In order to be simple and fast, the program structure of the stacked bed frame also makes the enterprise virtually impossible with less technical debt. The data is based on the factory label, all specifications, and stages of raw materials, semi-finished products, finished products and packaging. It will be assigned numbers and labeled for identification. Common label contents include: serial number, part number, model number, date, quantity and other text and barcode. The date range of this research is from the fourth quarter of 2018 to the second quarter of 2020, the data is established by the company in recent years by sign-off process into the information-based transfer process, but the variable field has been revised. It is confirmed that there are more than 30,000 effective documents that

have been applied to production line. The data retains historical version data, since changes and errors are the norm in the factory industry, after evaluation, these data are retained but not excluded, but need added to facilitate identification and influence the weight of the data through the characteristics, thereby obtaining correct results. The contributions of this paper can improve the problem with reducing production line shutdown and improve factory productivity. The accuracy rate of the artificial intelligence model can be increased to 95%. The number of stoppages is reduced from 4 times to 1 time per month. Under full capacity, this assist the decision-making system can reduce loss cost.

The remainder of the paper is as following: Section 2 provides a review of the Relative work; Section 3 outlines the research method, describes the exploration problems in this article and proposes the research process and overall structure; Section 4 is the experimental results and analysis; Section 5 is the conclusion future work.

2. Relative Work

There are three common applications of artificial intelligence in the manufacturing industry: scheduling optimization, numerical monitoring, and image recognition. The purpose of scheduling optimization is mainly to balance the load of the production line, shorten the production cycle, and reduce costs; the purpose of numerical monitoring is to reduce defective products, control yield and reduce costs; image recognition is often used in optical inspection stations for the purpose of reducing defective products, control yield, and reducing costs. There are many literatures on the application of artificial intelligence models in the manufacturing industry to study how to control the yield, but most of the artificial intelligence-related research focuses on the production environment [18, 19,20,21]. Even if problems are found through the artificial intelligence model, most cases still need to be stopped for adjustment, resulting in loss of productivity. If errors can be found when setting parameters at an earlier stage, the number of stoppages can be reduced. Fig. 1 shows the flow of the research data approval form.

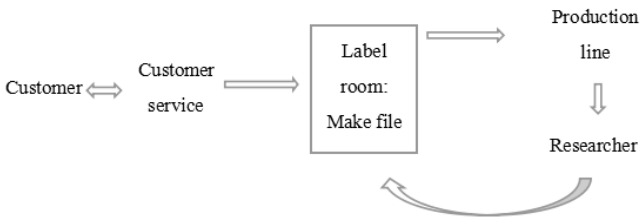


Fig. 1 Data flow

1. The customer provides the reference image file, style, printed content and other information of the label.
2. The Product Control Coordinator (PCC) obtains information through letters and

phone calls and fills in the application form.
3. The label room staff set the variable parameters corresponding to each item on the label.
4. The label room staff needs to ask information staff to assist in setting new variables.
5. The label room staff creates proofing label files based on information in the sign-off form.
6. After the approval is completed and effective, the label file is used to print the label during production on the production line.

2.1. Application of artificial intelligence in manufacturing

The relevant research on similar issues in the manufacturing industry is as follows: In 2020, Germany authors used KNN, random forest, and neural network model to solve the manpower problem required for the automatic optical inspection (AOI) result verification during the production of surface mount technology in the manufacturing industry. In order to solve the problem of the high cost of setting up inspection stations in the manufacturing industry, to replace the defect recognition system established by traditional computer vision with a deep learning model. Especially for the recognition of small objects, proposed a two-stage object detection algorithm [4, 13, 30, 32, 44]. To reduce the over-fitting situation, and finally use this automatic defect detection system to assist manual detection to reduce the missed detection rate and labor costs. Try to solve the problem of balancing the workload of the production line, reducing the number of workstations to reduce enterprise costs, and maximizing the work efficiency of the production line, using genetic algorithms, immune algorithms, and particle swarm algorithms, and proposed a new the coding method is discussed and compared, and the results show that the solution speed of the particle swarm algorithm is better than the other two algorithms, and the solution quality of the immune algorithm is better than the other two algorithms. In order to solve the problem of the deviation of the control parameters in semiconductor process production machine, which led to the deviation of the process, and caused the wafer yield to decrease, and even scrapped, A neural network combined with a failure detection and classification system to establish an early warning mechanism to ensure process yield and guaranteed production capacity [26, 31].

2.2. Artificial intelligence algorithm

The intelligence expressed by artificially manufactured machines can be called artificial intelligence, which is usually realized by computer programs and algorithms. It systematically learns from data and uses the learned knowledge to make predictions, thereby establishing expert systems and decision-making.

KNN. Nearest neighbor method, the classification of a target is determined by the classification of its K neighbors. It is explained by the spatial distribution. The concept is the same data higher the similarity, and closer the distribution. It is easy to understand in machine learning algorithms. The input of the model can be classification or

regression data, and the output is the category of the target. K is the number of nearest neighbors. If it is set too small, it will reduce the classification accuracy. If it is set too large, it may increase noise and affect the results. The model can be based on the neighbor distance gives different weights to optimize the results [7, 8, 9, 22, 39, 40]. When new data is available in this model, it can be added directly without retraining, and it is not sensitive to outliers; its disadvantage is needs to be re-calculated each time for the classification requires a large amount of memory, and it is sensitive to the local structure of the data. When the data is distributed unbalanced, the forecast is prone to deviation. Commonly used in text classification, pattern recognition, cluster analysis, multi-classification fields [42].

Decision tree. It is a tree model, where each node represents a feature, and each branch represents the possible attributes of the feature. Along the tree structure, the path is determined according to each feature attribute, and finally reaching the leaf node is the answer. The decision tree model can build classification trees, regression trees, and even both classification and regression trees that can be used. Basically, the decision tree has only a single output. When building a decision tree, it is necessary to determine the order of using features to create nodes based on the amount of variation as the division criterion. There are two common processing methods: use entropy to calculate the information gain, and subtract the information disorder before the division from the information after the division. If the decision tree model loses control and builds too many branches, it will be easy to overfit, so that all data has its own path, so it is necessary to restrict growth or even pruning. This model is easy to understand and implement, has a high degree of interpretability, does not need to do too much data pre-processing, and can process data and category data at the same time. Its disadvantage is that it is easy to overfit, and small changes in data are possible. When the data is unbalanced, the growth tends to have more numerical features, resulting in poor performance and neglecting the correlation between the attributes in the data [6].

Random forest. Random forest is composed of multiple decision trees. Because a single decision tree is easy overfitting, it has the characteristics of low deviation but high variance. Random forest uses random sampling and random selection of features to build multiple decision trees to solve this problem. Its output is determined by voting on the answers of all decision trees. The input of the model can be classification or regression data, and the output is the target category. The random sampling method of random forest is to use the method of retrieval and replacement. The same sample may be selected multiple times, or none of them may be selected at random. This model can evaluate the importance of features and has a fast training time. It can process high-dimensional data. If there is missing data, it can still maintain its accuracy. If the data is unbalanced, it has balance errors; its disadvantage is that the interpretation is poor. When there is a problem, it is impossible to make a prediction beyond the data range. If the data is noisy, it may still be over-fitting. It is often used in various classification and regression problems, or used in outlier detection, and also used in unsupervised learning classification problems [10, 12, 16, 28].

Neural Network. The concept of neural network is to simulate the function and structure of biological neural network through algorithms. Each neuron is composed of its input, excitation function, and output. The entire network is composed of input layer, hidden layer, and output layer. The input layer simulates numerous neurons receive huge non-linear input information. The hidden layer simulates the synaptic connection of

neurons and is responsible for transmitting the information to the corresponding position. The more complex non-linear relationships, the more it can lead to over-fitting [34, 36]. The output layer: The simulation information is processed through neural connections, and the model can learn by optimizing the weights of neurons in each layer and the connections between neurons. The input of the model can be classified or regression data, but the categorical data needs to be converted into numerical values through one-hot encoding. In the entire network, each neuron is connected to all the neurons in the next layer, and the neurons in the same layer. The elements are not connected to each other. The number of layers in the network and the number of neurons in each layer are determined by the complexity of the problem. It can be optimized and controlled during the parameter tuning stage. Usually, when the number of neurons is the same is a deep neural network. The performance is better than that of shallow neural networks. In addition to the development of deep learning research, it also extends more advanced methods such as recurrent neural networks and convolutional neural networks [27].

This model has self-learning ability, can learn the rules and patterns behind it from the data, and the learned knowledge is scattered throughout the neural network, so it has a certain fault tolerance, and a small part of the damage will not cause too much to the whole It can also adjust itself based on the learned results, and combined with the newly provided data; its disadvantage is that the learning speed is slow, and it is not explanatory. When the neural network is deeper and more complex, the huge amount of parameters requires a larger amount of data assists in training, otherwise it is easy to overfit. Because neural networks have infinite possibilities, it is difficult to find the best solution. In the process of tuning parameters, only a lot of attempts can be used to obtain better parameters. Commonly used in speech recognition, image recognition, recommend systems [33, 35].

3. Research Methods

The data used in the customized labeling and endorsement form of electronic manufacturer. The fields filled in the applicant are the input data of the artificial intelligence model. The variable setting parameters actually set in the label room are the artificial intelligence model. This experiment will build hardware specifications: processor CPU Xeon-E5-2650, memory DDR4 64 GB, operating system Windows 10, software based on open source software Python, and use the python library provided by OpenCV, and use Keras, Tensorflow, etc. as neural network development tools. Implemented in Windows and Linux systems.

3.1. Research Structure

As shown in Fig. 2, the research steps are mainly divided into three blocks: data processing, construct modeling, and optimized experiment. The purpose of data processing in the first block is to prepare the input data required for construct modeling, including data collection, sorting and screening. Data pre-processing that the model can

operate normally and correctly. Feature construction and selection to clarify the importance of features. The purpose of construct modeling, in order to find a model that is more suitable for the data type, seven models that are well-known in the performance of categorical data are selected and compared with each other. Optimized experiment based on the results of the previous block, after finding a better model, try to optimize further and obtain better results. This part is divided into three types of experiments, feature processing experiments, circuit training experiments, and parameters experiment. Finally, the stage results of each step can be used as fixed parameter for other steps, and optimize other steps.

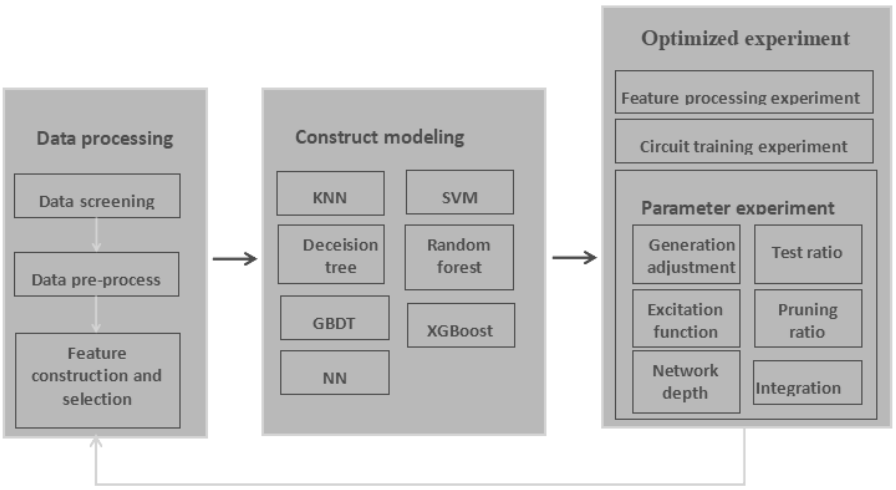


Fig. 2. Research architecture

3.2. Data Collection, Sorting and Screening

The data of this research is stored in the company's internal database. According to the approval form process, three types of data sheets are designed to store. One is the approval form data, which records the contents filled in by each unit in the transmission and can be based on the site authority of the approval process modification; Two is the historical version data that has been validated after the approval is completed, and it is provided for confirmation by the customer and the inspection unit, which is also the basis for the next revision; the third is the data actually used by the production line after it takes effect, which is extremely sensitive data. Once modified, it will be directly modified affect the production line. The data used in this study falls into the second category: historical version data that has been validated and entered into force. As the information in subscribe, there will be many blank fields that have not been filled in, and there may be errors that have not been finalized. On the other hand, the data actually used by the production line is only the latest version of the data, the data is not

continuous, and there is no past data, resulting in too little data volume, it is difficult to find out the setting rules. The historical version data is only recorded when the production line takes effect, not only can ensure the accuracy of the data, but also can be traced back to the past. Using this data to train an artificial intelligence model, it is expected that the model can learn the rules behind each customized label over time and version evolution. The data are stored independently in five factories under the same structure. Different factories may face different customers, fill in forms and set up personnel, and the proportion of data varies greatly. Compared with the scale of data 3 to 26 times, so we finally decided to use only the largest plant area data for research.

The data comes from 4400 signed forms, but because only the second type of signed forms are used, the number of valid forms is actually only 3800, which is about the total number of forms. The source of these forms comes from 2800 different material numbers, but only 2500 material numbers have gone through the sign-off process. Each form contains one or more different label types, and all the used label types are 24 types, only 23 types are actually applied to the production line. In practice, there are 6 types that are more commonly used: 1. CB_SN, semi-finished product customer label, mostly a two-dimensional code, used to record the serial number of the semi-finished product stage; 2. SN, production line input the serial number label, mostly one-dimensional code, to record the production serial number in the factory; 3. FCC, the product label on the host, mostly one-dimensional code, record the product serial number; 4. BOX, the label on the color box, mostly one-dimensional code, record the serial number of the color box; 5. CARTON, the label on the outer box, mostly composite bar code, including the outer box serial number, product specifications, quantity, content serial number list; 6. PALLET, pallet label, mostly composite bar code, including pallet serial number, product specification, quantity, content serial number list. The other types are usually an extra label at the same position to display additional content or special specifications. Because the amount of data is small or too special, this research will skip these data and not adopt it, and use six types is the main research data.

A label type contains one or more different variables. Among 3800 completed forms contains 41000 variable data. However, the variable settings have been revised. The old variable names have been discarded and no longer used. There are only 24500 data on the new system variables. There are only 22,600 items left. The field that may be incorrectly set by the setting personnel is the variable parameter field. It will list as the final prediction target of the artificial intelligence model. Under all label types, there are 580 answers in the variable field. Focusing on six types of data, there are 460 answers after screening. Among all the types, the answers to some variables are the same. After taking the intersection, there are 270 types of answers, while the six types of intersections are left with 260 types of answers. The unpredictable unknown is regarded as one answer added, namely the largest output layer dimension of the artificial intelligence model. It use the 580 variable types, the number of variable types and the amount of data of each type are not even. Sort is according to the number of variable types: FCC> CARTON> BOX> SN> PALLET> CB_SN; sort is according to the amount of data: CARTON > BOX> FCC> PALLET> CB_SN> SN. The more types of variables, the more complicated the rules behind, the higher the error rate may be included; if the amount of data is less, the more difficult it is to accumulate experience and learn the rules, and the error rate will also increase. Looking at these two elements together, the ratio of the amount of available data divided by the number indicates how

much data can be used in training for each answer. Sort is according to this ratio: CB_SN> CARTON> BOX> PALLET> the sum of the six categories> All> SN> FCC, it can be expected that CB_SN has the most learning materials and is the easiest to learn; FCC has the least learning materials and is the least easy to learn.

All types of data, the utilization rate of variable types is extremely uneven. The usage rate of serial number, quantity and date is particularly high. Among the 583 variable types, the top two together account for 20%, the top four together account for 30%, and the top six together account for 40%. The top eight together accounted for 50% of the ratio. For the overall accuracy rate, it uses the sum of the proportions of several commonly used variables as one of the control groups, and then compares it with the trained artificial intelligence model.

3.3 Data pre-processing

In the process information, it is inevitable that some data is incomplete. It may be that the data table structure design is not perfect at the beginning, resulting in some data not being collected at the initial stage; or the personnel are not rigorously operated, causing the data be misplanted; or the data is improperly maintained and other modifications are made. Data is accidentally moved by mistake, causing data abnormalities; it may also cause abnormal system instability and data damage. The situation and processing methods of incomplete data are as follows: When there are missing values or missing data, the common processing method is to discard or supplement the value. Because the amount of data is not large, the available data should not be discarded as much as possible.

In some models, the category field needs to be converted by one hot encoding before it can be used. Most of the data fields are category data, and one category field needs to be converted into multiple values using one hot encoding. The fields of 0 or 1 are then passed as input to the model for learning. The text field is usually processed after cutting keywords or conversion vectors. However, some field data may contain valuable hidden data. After combining professional knowledge and experience, the hidden information of the data can be integrated and extended. The data implies some information that cannot be seen directly on the surface or item number. The first two codes of the data represent the production stage of the factory for the item number that extract into a new field may help the model.

3.3. Feature Construction and Selection

The original data contains multiple features, but not all features are necessary. Too many features may interfere with the learning of the model, and there may even be doubts about dimensional explosion. Reducing unnecessary features can reduce the calculation of the model. In practice, it is interpretable to convince the boss or customer. Many reasons show the necessity of analyzing the importance of features. After analyzing the importance of data features, it can be used to verify the correctness of the model. If the prediction is inaccurate, this judgment basis can more quickly clarify the problem and improve the model. For important features, in addition to optimizing the existing model,

it also collecting follow-up data, it can enhance the user's proper use of important features. At this stage, it tries to clarify the importance of each feature, find out the key features in different feature fields, and confirm whether each feature is as expected to help the model predict the correct answer. In addition to preliminary screening based on experience and professional knowledge, it uses the random forest model to explore the importance of features. Random forest judges the importance of features by calculating the contribution of each feature on each tree, and then take the average value and draw a graph.

3.4. Common Model Modeling and Comparison

The data field is mainly based on categories. When you are not sure which model is more suitable for this data type, try and compare several common models that perform better in classification. The data that has been pre-processed before is modeled, and the effectiveness is evaluated. As shown in Table 1, the advantages and disadvantages of the artificial intelligence model are sorted out.

Table 1 Comparison of advantages and disadvantages of artificial intelligence models

Model	Advantages	Disadvantages
KNN	Information can be added directly without retraining. It is fast and insensitive to outliers.	Each classification needs recalculate. The memory demand is large. When data distribution is unbalanced, the forecast is easy to be biased.
SVM	Strong generalization ability and is easy to understand. Part of the data can be used to make hyperplane decisions. It works well for processing high-dimensional data.	Poor performance and sensitive to missing data. There is no universal solution to nonlinear problems. The explanation of the kernel function is not high.
Decision tree	Easy to understand and implement, and has high degree interpretability. No need to do much data pre-processing, can process data and category information at the same time	It is easy to overfit, the result of data changes is unstable, and the performance is poor when the data is unbalanced. Ignore the correlation between the attributes in the data.
Random forest	Not easy to overfit, training speed is fast, The importance of features can be assessed with high accuracy. Can deal with missing data, and unbalanced data.	The interpretation is poor, when dealing with regression problems. It is impossible to make predictions beyond the scope of the data. If the data is noisy, it may still overfit.
GBDT	Can prevent overfitting and does not require complex features, characteristic processing, non-linear transformation, can process linear or non-linear data.	The computational complexity is high. Failure to parallelize is time-consuming. Not suitable for sparse high-dimensional data (Liu et al., 2018).
XGBoost	Prevents over-fitting, nodes at the same level can be parallel processing, can handle sparse data.	When a node splits, it is necessary to traverse the data set. Use twice as much memory (Jidong et al., 2018).
NN	Flexible, it has fault tolerance and self-adjustment ability.	It is not explanatory and computationally intensive. A lot of information need tuning and requires more trial and error.

3.5. Optimization model

Feature processing. Through more complete data pre-processing, the model learning effect can be better, mainly for the processing of non-required fields and fuzzy data. Use all the models to try to learn the modeling for the data before and after processing, and

then evaluate and compare benefit. Complementary value method: SN_NAME is required, but may be an alias. For fuzzy processing method, there is no mandatory and standardized way to fill in these two fields, and there is no restriction on symbols and capitalization. There may be several types of filling for the same object. Therefore, after removing the symbols in the two fields, only letters, numbers and then the letters are converted to uppercase, so that the fuzzy approximate data can be integrated into consistent data.

Loop experiment. After testing the neural network, it was found that the data was too inconsistent with the actual application. After analysis, it was found that the data has time characteristics. If the training set and the test set are randomly selected, there may be future data of the test data in the training set. The answer before verification may cause the model to get out of control, overfitting, and failing to learn correctly. Therefore, the experiment was designed to clarify the actual situation based on the test set segmentation method. During the analysis and evaluation, it was found that the material numbers in the data were unevenly distributed in the entry action. Due to the time characteristics, the test data could not be randomly selected. In this case, the data diversity in the test set was low. As shown in Fig. 3, in order to increase the diversity of the test set, partial data is used section by section through loops, so that more diverse data can have the opportunity to act as test data.

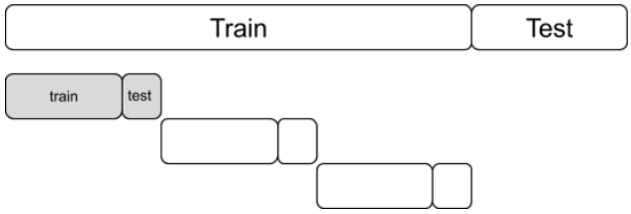


Fig. 3. Schematic diagram of loop training

As shown in Fig. 4, in order to increase diversity, different methods of using partial data are designed so that more data can have the opportunity to act as test data. In order to avoid the use of partial data and too little data, the models are all memorizing answers, this experiment uses all types together.

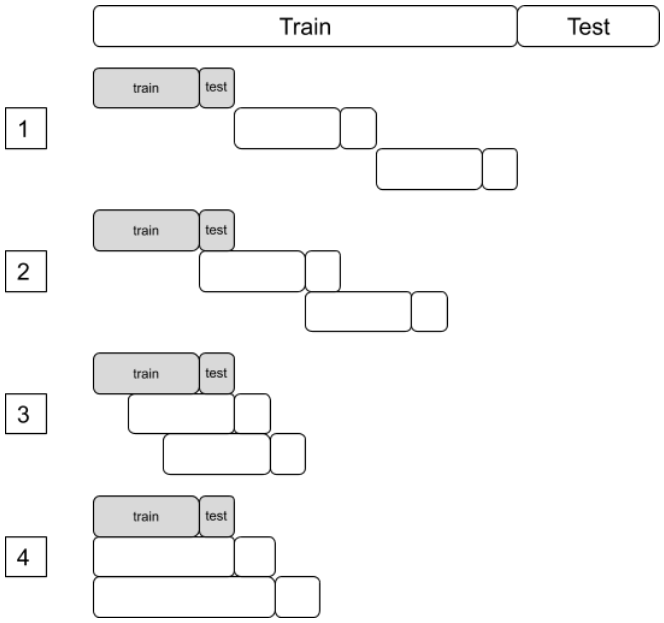


Fig. 4. Schematic diagram of partial data usage of loop training

Method one is that all data is used only once regardless of training or testing; method two is that all data will be put into the training set once; method three is that all data will be used once in the test set; method four is that all data will be used in the test set once, the used data as the next training set, and then take a certain percentage of the new data as the test set.

Parameter experiment. The model needs to adjust the parameters according to the data type to find a better parameter set to perform more ideally, so that the model has a better performance. Based on the results of modeling comparison, this part is for neural network. First, for the generation parameters, observe the learning curve of the data to find out the generation parameters that are suitable for the data type; then adjust the test set proportions and observe the changes in accuracy to determine the better test set proportion parameters; then adjust the excitation function, the time factor is also taken into consideration to determine the better excitation function parameters; in order to prevent over-fitting, observe the changes in the accuracy rate to find the better pruning ratio parameters; to ensure neural network model has enough neuron connections to learn the data rules completely, do experiments to confirm the best network depth; finally, integrate the parameters obtained at each stage to verify whether the neural network model progresses as expected.

4. Results and Analysis

The research results and experimental results are presented and analyzed in the following. Although the results of some experiments are not very significant, as long as each link can be optimized, it will eventually bring great improvement.

4.1. Feature Construction and Selection

Using the random forest model, it is found that the features are not the original data field names, and the importance proportion is also very low. Most of the researched data are categorical fields, which have been converted into multiple fields after one hot encoding. The importance of each original feature can be added the importance value of each derived field based on the original field. That SN_NAME has the highest importance, which is higher than SN_TITLE. The importance of ACTION_TYPE is the lowest.

4.2. Common Model Modeling and Comparison

The control group in this experiment uses practical statistical data in company. Each time the form is entered, the average change content is 20%. If the entire form is sent directly without modification, the accuracy rate is 80%.

Table 2. Experimental results of different artificial intelligence model

	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
Blind guess	0.11	0.17	0.25	0.21	0.15	0.18	0.22	0.01
Base				0.80				
KNN	0.85	0.91	0.85	0.92	0.76	0.84	0.72	0.87
SVM	0.26	0.38	0.40	0.83	0.28	0.58	0.28	0.44
Decision tree	0.19	0.29	0.39	0.74	0.25	0.57	0.45	0.47
Random forest	0.87	0.93	0.88	0.94	0.85	0.87	0.89	0.89
GBDT	0.764	0.86	0.80	0.92	0.73	0.75	0.69	0.78
XGBoost	0.84	0.92	0.84	0.95	0.78	0.85	0.78	0.84
NN	0.88	0.89	0.88	0.93	0.84	0.89	0.80	0.86

As shown in Table 2, the horizontal axis of this table is of various types: All means all types; others is the remaining data after excluding the six types. The vertical axis is the control group and various models. Looking at the vertical axis, that regardless of the model, the accuracy of CB_SN is very high. It can be found that the target field of CB_SN has the least type, but in addition to looking at the target field type, attention should also be paid to the amount of data. Divide the number of target fields of each type by the ratio of the amount of data to sort to get: CB_SN > CARTON> BOX> PALLET> All > SN> FCC. It founds that the higher the data ratio, the more data available for training, the higher the accuracy rate, because there is still a problem of sparseness between the amount of data and the target field that needs to be considered.

The CARTON data ratio is the 2nd, and the BOX data ratio is the 3rd. However, the performance of these two types in SVM and decision tree is worse than that of the 4th PALLET in the data ratio. Looking at the horizontal axis, the accuracy of KNN, random forest, XGBoost, NN and other models exceeds the control group, reaching the reference standard, and the performance is good.

4.3. Evaluation of the Effectiveness of Data Pre-processing

In the previous step to clarify the importance of features, it shows that SN_TITLE and SN_NAME are the more important fields, and these two fields have room for further optimization. After compensation and processing of fuzzy data, each artificial intelligence model is retrained and observed accuracy. Complementary value method: SN_NAME is required, but may be an alias; SN_TITLE is not required and may be empty. Filling in the empty SN_TITLE into the data to the SN_NAME field. Fuzzy processing method: There is no mandatory and standardized way to fill in these two fields, as long as users of each station can understand it, and there is no restriction on symbols and capitalization. Depending on the user's habits, there may be several types of filling for the same object. Therefore, after removing the symbols in the two fields, only letters, numbers, and Chinese are left, and then converted to uppercase, so that the fuzzy approximate data can be integrated into consistent data.

As shown in Table 3, the horizontal axis of this table is various types, the vertical axis is the control group and various models, and the data column is the accuracy rate change value. It shows that SVM and decision tree have increased significantly, and the accuracy rate has increased; as for KNN, random forest, GBDT, XGBoost and NN, the impact is less significant. Some models may have already learned the upper limit on the existing data and features, so there is not so much for optimization, or some models such as random forest can deal with the problem of missing values or fuzzy data.

4.4. Evaluation of the Effectiveness of Loop Training

The experiment tries to optimize the neural network model such as the best overall performance, explains the purpose of the loop test, analyzes and discusses the experimental results.

Comparison of effectiveness evaluation of loop training. 80% of the data is used for training, and only 20% is used for verification. Because the data field information is mostly categorical data, and the interval time of each material number is inconsistent, resulting in uneven data distribution. Due to time characteristics, the test data cannot be randomly selected. In this case, the data diversity in the test set is low. In order to solve this problem, this experiment was designed in order to increase the diversity of the test set, partial data is used section by section through a loop, so that more data can have the opportunity to act as the test data.

Table 3. Experimental results after data processing

	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
KNN	+0.01	-0.01	+0.01	+0.01	+0.01	+0.02	+0.02	+0.01
SVM	+0.21	+0.18	+0.14	+0.09	+0.14	+0.02	+0.09	+0.05
Decision tree	+0.09	+0.09	+0.06	+0.11	+0.18	-0.01	+0.01	+0.02
Random forest	+0.00	-0.03	-0.01	+0.00	+0.02	+0.02	+0.01	+0.03
GBDT	+0.00	+0.05	+0.02	+0.01	+0.02	+0.01	+0.05	-0.01
XGBoost	+0.03	+0.01	+0.01	-0.02	-0.03	-0.00	-0.01	-0.01
NN	+0.00	+0.01	-0.01	+0.01	+0.00	+0.00	+0.03	+0.01

As shown in Table 4, the horizontal axis of this table is various types, and the vertical axis is the number of cycles. In this experiment, the total data volume of the test set is the same as that of the control group. It can be proved that only the test set is dispersed and increased diversity. It can indeed improve results. Only the results of the SN type are lower than the control group, which may be related to the fact that most of the data types are special cases. In table 4 that the best result has no significant relationship with the number of cycles, and it is speculated that it should be more related to the diversity of the test set distribution.

Table 4. Results of the loop training experiment

	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
NN Base	0.89	0.90	0.86	0.94	0.84	0.89	0.84	0.87
Roll 2	0.91	0.93	0.91	0.92	0.83	0.90	0.79	0.86
Roll 3	0.92	0.95	0.93	0.94	0.82	0.94	0.78	0.84
Roll 4	0.91	0.97	0.92	0.94	0.83	0.90	0.77	0.89
Roll 5	0.90	0.97	0.93	0.94	0.80	0.88	0.71	0.87
Roll 6	0.90	0.95	0.96	0.93	0.87	0.86	0.71	0.84
Roll 7	0.89	0.94	0.94	0.92	0.89	0.84	0.83	0.85
Roll 8	0.91	0.95	0.95	0.88	0.83	0.87	0.78	0.80
Roll 9	0.92	0.93	0.91	0.84	0.91	0.80	0.78	0.78
Roll 10	0.91	0.93	0.94	0.88	0.84	0.86	0.70	0.76

Comparison of effectiveness evaluation of loop training. In order to increase diversity, design different methods of using partial data, so that more data can have the opportunity to act as test data. In order to avoid the use of partial data, the amount of data is too small, resulting in the model is memorizing answers, and therefor experiment only uses all types to carry out. The control group is the basic neural network model and method one of the previous experiment. Method one is that all data is used only once regardless of training or testing; method two is that all data will be put into the training set once; method three is that all data will be used once in the test set; method four is that all data will be used in the test set once, the data used each time is used as the next training set, and a certain percentage of the new data is taken as the test set.

As shown in Table 5, the horizontal axis of this table is the amount of data for each training of methods one to three, the control group and various methods, and the vertical axis is the proportion of the amount of data used for each cycle training.

Table 5. Advanced experimental results of loop training

Data volume	Training set	Test set	NN Base	Method 1 Base	Method 2	Method3	Method4
1/2	9796	2450		0.91	0.91	0.92	0.89
1/3	6531	1633		0.92	0.93	0.92	0.84
1/4	4898	1225		0.91	0.94	0.94	0.86
1/5	3918	980		0.90	0.93	0.92	0.86
1/6	3265	817		0.90	0.95	0.95	0.89
1/7	2798	700		0.89	0.89	0.94	0.76
1/8	2448	613	0.89	0.91	0.94	0.92	0.79
1/9	2176	545		0.92	0.91	0.91	0.79
1/10	1959	490		0.91	0.92	0.92	0.79

From table 5 shows that the results of method two and method three are better than those of the control group, but only looking at the data, there is no obvious advantage or disadvantage between method two and method three. The result is generally lower than that of the control group. After analysis, it is found that method 2 is relatively normal, and method 3 is suspected of overfitting. Therefore, method 2 is selected, and the amount of data is 1/6 which is the better parameter for the experiment.

4.5. Evaluation of the effectiveness of model optimization

The experiments are aimed at optimizing neural network models such as the best overall performance, and discuss and compare the results after sorting out the experimental results.

Comparison of effectiveness evaluation of adjusting the test set proportion. This experiment adjusts the test set ratio and observes the changes in accuracy. As shown in Table 6, due to the numerous combinations of loop training methods and data volume, this experiment only takes the methods and data volume parameter sets that perform well in the previous loop training experiments for further attempts. The horizontal axis is the control group and cyclic training parameters, the first parameter is the method, and the second parameter is the data volume denominator; the vertical axis is the test set ratio, under NN Base, 0.1> 0.2> 0.15> 0.3> 0.25; Under Roll(1,9), 0.1> 0.15> 0.2> 0.25> 0.3; under Roll(2,6), 0.1> 0.2> 0.15> 0.3> 0.25; under Roll(3,7), 0.2> 0.25> 0.15 = 0.3> 0.1. The overall effect is better with a result of 0.2. At first glance, the smaller the test ratio, the higher the accuracy rate. However, after analysis, it is obvious that the high accuracy rate is illusion when the test ratio is small. As the test ratio is less, the diversity of test data is lower. The more difficult it is to verify the correctness of the data; if the test ratio is too high, it will in turn lead to too low diversity of training data, resulting in a sharp drop in accuracy, so 0.2 is the better test ratio in NN based method.

Table 6. Test proportions in the loop training experiment result

Test ratio	NN Base	Roll(1,9)	Roll(2,6)	Roll(3,7)
0.1	0.91	0.96	0.97	0.83
0.15	0.89	0.92	0.93	0.92
0.2	0.89	0.92	0.95	0.96
0.25	0.80	0.89	0.93	0.94
0.3	0.81	0.87	0.93	0.92

Comparison of effectiveness evaluation of adjusting trigger function. This experiment adjusts the excitation function and observes the changes in accuracy.

Table 7. Excitation function experiment results

Parameter	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
softmax	0.12	0.12	0.23	0.24	0.12	0.17	0.18	0.14
sigmoid	0.88	0.88	0.87	0.92	0.79	0.87	0.82	0.83
elu	0.88	0.89	0.87	0.94	0.83	0.89	0.83	0.85
relu	0.89	0.90	0.86	0.94	0.84	0.89	0.84	0.87
selu	0.89	0.91	0.88	0.95	0.85	0.89	0.82	0.87

As shown in Table 7, the horizontal axis of this table is each type, and the vertical axis is the excitation function parameter. Under the all type, relu> selu> elu> sigmoid> softmax; under the BOX type, selu> relu> elu> sigmoid> softmax; CARTON Under the type, selu> elu> sigmoid> relu> softmax; under the CB_SN type, selu> elu> relu> sigmoid> softmax; under the FCC type, selu> relu> elu> sigmoid> softmax; under the PALLET type, relu> selu = elu > sigmoid> softmax; for SN type, relu> elu> sigmoid> selu> softmax; for Others type, selu> relu> elu> sigmoid> softmax. On the whole, relu ≒ selu> elu> sigmoid> softmax, but the difference between relu, selu, elu, and sigmoid is not big, the difference is less than 1% for the All types; the difference is about 4% for the BOX type; the difference is about 4% for the CARTON type. the difference is about 2.5% for the CB_SN type; the difference is about 6% for the FCC type; the difference is about 2% for the PALLET type; the difference is about 1.5% for the SN type; the difference is about 4% for the others type. Since the execution time of relu is faster, the time difference is about 20%, so relu is the better setting.

Adjusting the depth of the network effectiveness evaluation comparison. The data in this study has been transformed or the input layer has a dimension of about 1,000, and the output layer has a dimension of about 256. This experiment adjusts the number of hidden layers to observe the changes in accuracy. Due to the limited amount of data in this study, if the depth of the network is too deep, the parameters will be too large and over-fitting. Therefore, this experiment only tested three hidden layers at most.

Table 8. Network depth experiment results table

layer	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
1	0.88	0.88	0.88	0.92	0.79	0.89	0.83	0.84
2	0.89	0.90	0.87	0.94	0.84	0.89	0.84	0.87
3	0.89	0.90	0.87	0.93	0.83	0.89	0.82	0.85

As shown in Table 8, the horizontal axis of this table is each type, and the vertical

axis is the number of hidden layers. Under the All type, 2> 3> 1; under the BOX type, 3> 2> 1; under the CARTON type, 1> 3> 2; Under CB_SN type, 3> 2> 1; Under FCC type, 2> 3> 1; Under PALLET type, 3> 2> 1; Under SN type, 2> 1> 3; Under Others type, 2> 3> 1. On the whole, the difference between the results of each network depth is not big, the difference is less than 1% under the All type; the difference is about 2% under the BOX type; the difference is approximately 2% under the CARTON type; the difference is approximately 3.5% under the CB_SN type; The difference is about 5.5% for FCC type; about 1% for PALLET type; about 2% for SN type; and about 3% for others type. In order to solve the problem of over complexity, the deeper the neural network is the better, but if it is too deep, it may cause overfitting. From the table that under different types, the best and the worst are not much different. The neural network model can learn well under the basic structure, so it is impossible to optimize too much through this parameter. The overall effect is the result of second hidden layers is better. Due to the numerous combinations of loop training methods and data volume, this experiment only takes the methods and data volume parameter sets that perform well in the previous loop training experiments for further attempts. As shown in Table 9, the horizontal axis of this table is the control group and loop training parameters. The first parameter is the method and the second parameter is the denominator of the data volume; the vertical axis is the number of hidden layers, under NN Base, 2> 3> 1; under Roll (1, 9), 3> 2> 1; under Roll (2, 6), 2> 1> 3; under Roll (3, 7), 1> 2> 3; the overall effect is better with second hidden layers, but the difference is very small. The difference between the results of each network depth is not big, the difference is about 0.5% under Roll (1, 9); the difference is about 0.5% under Roll (2, 6); the difference is about 0.5% under Roll (3, 7). In order to solve overly complex problems, the deeper the neural network depth is the better, but too deep may cause over-fitting. Under different types, the difference between the best and the worst is only 0.5% are no obviousdifferent. The neural network model can learn well under the infrastructure even after it is trained in a loop. Therefore, it is impossible to optimize too much through this parameter, and the overall effect is better. The results of second hidden layers is better.

Table 9. Network depth in the loop training experiment results table

Layer	NN Base	Roll(1,9)	Roll(2,6)	Roll(3,7)
1	0.88	0.90	0.94	0.96
2	0.90	0.91	0.95	0.95
3	0.89	0.92	0.93	0.94

5. Conclusion

This research collects the fill-in content of the approval form of customized labels in the manufacturing industry to predict production line labels. In the study, such as random forests and neural networks were used for training and prediction. The experiments results of this research show that the random forest and the neural network method is better performance in AI method. The auxiliary decision-making system established by the better trained model can reduce the error rate in practice and maintain the productivity of the factory. Through the results of this research, it shows that among the

data features, the label alias is the most important and has the greatest impact. The contributions of this paper:

1. According to statistics, the production line before tuning, the accuracy rate of new recruits is only about 80%, after use the best of artificial intelligence models, can reach to 95%.
2. The number of stoppages is reduced from 4 times to 1 times per month. In the case of full capacity, this research can assists decision making system to reduce loss cost.
3. The production line labels decision making was established, can enhance the operating experience and improve work efficiency, reduce error rate and increase factory productivity.

This artificial intelligence module is based on the factory approval form and parameter settings. There are many systems in the company that involve verification and parameter setting. If this research can be extended to other application scenarios in the factory, it will be optimize and overall improvement. Future work, if the filling standard can be promoted, additional valuable implicit information can be obtained from this study when conducting research in the future. It also can assist artificial intelligence model for training and learning too. Limitation is setting errors may cause abnormalities during production and reducing factory productivity and bring losses to the enterprise.

References

1. Bai Y, Li C, Sun Z, Chen H, (2017) Deep neural network for manufacturing quality prediction," 2017 Prognostics and System Health Management Conference. Harbin, 2017, pp. 1-5, doi: 10.1109/PHM.2017.8079165.
2. Cheng, H., Wu, L., Li, R., et al. (2021) Data recovery in wireless sensor networks based on attribute correlation and extremely randomized trees[J]. Journal of Ambient Intelligence and Humanized Computing, 12(1): 245-259.
3. Cheng, Y., Jiang, H., Wang, F., et al. (2019) Using High-Bandwidth Networks Efficiently for Fast Graph Computation[J]. IEEE Transactions on Parallel and Distributed Systems, 30(5): 1170-1183.
4. Dai, Y., Wang, S., Chen, X., et al. (2020) Generative adversarial networks based on Wasserstein distance for knowledge graph embeddings [J]. Knowledge-Based Systems, 190: 105165.
5. Ding Z, Peng W, Yan Q, Lin F (2019) Research on Intelligent Manufacturing System of Sustainable Development," 2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, 2019, pp. 657-660, doi: 10.1109/WCMEIM48965.2019.00139.
6. Elaidi H, Elhaddar Y, Benabbou Z, Abbar H (2018) An idea of a clustering algorithm using support vector machines based on binary decision tree," 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, 2018, pp. 1-5, doi: 10.1109/ISACV.2018.8354024.
7. Fu, Y. G., Huang, H. Y., Guan. Y., et al. (2021b) EBRB cascade classifier for imbalanced data via rule weight updating [J]. Knowledge-Based Systems, 223: 107010.
8. Fu, Y. G., Ye, J. F., Yin, Z. F., et al. (2021a) Construction of EBRB classifier for imbalanced data based on Fuzzy C-Means clustering[J]. Knowledge-Based Systems, 234: 107590.
9. Fu, Y. G., Zhuang, J. H., Chen. Y, P., et al. (2020) A framework for optimizing extended belief rule base systems with improved Ball trees[J]. Knowledge-Based Systems, 210:

106484.
10. Ghosh S, Banerjee C (2020) A Predictive Analysis Model of Customer Purchase Behavior using Modified Random Forest Algorithm in Cloud Environment. 2020 IEEE 1st International Conference for Convergence in Engineering (ICCE).
11. Jidong L, Ran Z (2018) Dynamic Weighting Multi Factor Stock Selection Strategy Based on XGboost Machine Learning Algorithm," 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), Chongqing, China, 2018, pp. 868-872, doi: 10.1109/IICSPI.2018.8690416.
12. Kou L, Liu C, Cai G, Zhang Z, Li X, Yuan Q (2020) Fault Diagnosis for Power Converters based on Random Forests and Feature Transformation. 2020 IEEE 9th International Power Electronics and Motion Control Conference.
13. Li, X. Y., Lin, W., Liu, X., et al. (2022) Completely Independent Spanning Trees on BCCC Data Center Networks with an Application to Fault-Tolerant Routing[J]. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 33(8): 1939-1952.
14. Liu A, Liu X, Yu H, Zhang C, Liu Q, Tao D (2021) Training Robust Deep Neural Networks via Adversarial Noise Propagation. IEEE Transactions on Image Processing. 30.
15. Liu S, Cui Y, Ma Y, Liu P (2018) Short-term Load Forecasting Based on GBDT Combinatorial Optimization," 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, 2018, pp. 1-5, doi: 10.1109/EI2.2018.8582108.
16. Liu Y, Liu L, Gao Y, Yang L (2019) An Improved Random Forest Algorithm Based on Attribute Compatibility," 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 2019, pp. 2558-2561, doi: 10.1109/ITNEC.2019.8729146.
17. Liu Y, Zhao Y, Tao L, Zhao K, Li K (2018b) The Application of Digital Flexible Intelligent Manufacturing System in Machine Manufacturing Industry," 2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Tianjin, China, 2018, pp. 664-668, doi: 10.1109/CYBER.2018.8688303.
18. Liu, G., Chen, X., Zhou, R., et al. (2021) Social learning discrete Particle Swarm Optimization based two-stage X-routing for IC design under Intelligent Edge Computing architecture[J]. Applied Soft Computing. 10, 107215.
19. Liu, G., Chen, Z., Zhuan,g Z., et al. (2020a) A unified algorithm based on HTS and self-adapting PSO for the construction of octagonal and rectilinear SMT[J]. Soft Computing, 24(6): 3943-3961.
20. Liu, G., Zhang, X., Guo, W., et al. (2022a) Timing-Aware Layer Assignment for Advanced Process Technologies Considering Via Pillars. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 41(6): 1957-1970.
21. Liu, G., Zhu, Y., Xu, S., et al. (2022b) PSO-Based Power-Driven X-Routing Algorithm in Semiconductor Design for Predictive Intelligence of IoT Applications. Applied Soft Computing, 114: 108114.
22. Liu, N., Pan, J. Sun, C., et al. (2020c) An efficient surrogate-assisted quasi-affine transformation evolutionary algorithm for expensive optimization problems[J]. Knowledge-Based Systems, 209: 106418.
23. Lu, Z., Liu, G., and Wang, S. (2020) Sparse neighbor constrained co-clustering via category consistency learning[J]. Knowledge-Based Systems, 201, 105987.
24. Muhammad Y, Cong P, Lu H, Fan Y (2010) MES development and significant applications in manufacturing -A review," 2010 2nd International Conference on Education Technology and Computer, Shanghai, 2010, pp.97-101, doi: 10.1109/ICETC.2010.5530040.
25. Ostrowski D (2018) Artificial Intelligence with Big Data," 2018 First International Conference on Artificial Intelligence for Industries (AI4I), Laguna Hills, CA, USA, 2018, pp.125-126, doi: 10.1109/AI4I.2018.8665678.
26. Ozdemir R, Koc M (2019) A Quality Control Application on a Smart Factory Prototype

Using Deep Learning Methods," 2019 IEEE 14th International Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 2019, pp. 46-49, doi: 10.1109/STC-CSIT.2019.8929734.
27. Pauli P, Koch A, Berberich J, Kohler P, Allgöwer F (2022) Training Robust Neural Networks Using Lipschitz Bounds. IEEE Control Systems Letters. Volume: 6.
28. Prihatno A, Nurcahyanto H, Jang Y. M (2021) Predictive Maintenance of Relative Humidity Using Random Forest Method. 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC).
29. Shen, S., Yang, Y., Liu, X. (2021) Toward data privacy preservation with ciphertext update and key rotation for IoT[J]. Concurrency and Computation: Practice and Experience, e6729.
30. Song H, Huang G, Chauvel F, Xiong Y, Hu Z, Sun Y, Mei H (2011) Supporting runtime software architecture: A bidirectional-transformation-based approach. Journal of Systems and Software, 84(5): 711-723.
31. Thielen N, Werner D, Schmidt K, Seidel R(2020) Reinhardt and J. Franke, "A Machine Learning Based Approach to Detect False Calls in SMT Manufacturing," 2020 43rd International Spring Seminar on Electronics Technology (ISSE), Demanovska Valley, Slovakia, 2020, pp. 1-6, doi: 10.1109/ISSE49702.2020.9121044.
32. Wang, S., Wang, Z., Lim, K. L., et al. (2021) Seeded random walk for multi-view semi-supervised classification [J]. Knowledge-Based Systems, 222: 107016.
33. Yang Y, Guan Z, Wan Z, et al. (2021) PriScore: blockchain-based self-tallying election system supporting score voting[J]. IEEE Transactions on Information Forensics and Security, 16: 4705-4720.
34. Yang Y, Liu X, Deng R. (2018a) Expressive Query over Outsourced Encrypted Data. Information Sciences, 442: 33-53.
35. Yang Y, Liu X, Guo W, et al. (2020) Multimedia access control with secure provenance in fog-cloud computing networks. Multimedia Tools and Applications, 79(15): 10701-10716.
36. Yang Y, Zheng X, Chang V, et al. (2018b) Lattice assumption based fuzzy information retrieval scheme support multi-user for secure multimedia cloud. Multimedia Tools and Applications, 77(8): 9927-9941.
37. Yu Z, Zheng X, Huang F, et al. (2021) A framework based on sparse representation model for time series prediction in smart city[J]. Frontiers of Computer Science, 15(1): 1-13.
38. Zhang Y, Huang G, Liu X, Zhang W, Mei H, Yang S (2012) Refactoring android Java code for on-demand computation offloading. ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications.
39. Zhang, H., Li, J. L., Liu, X. M., et al. (2021a) Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection[J]. Future Generation Computer Systems, 122: 130-143.
40. Zhang, Y., Lu, Z., and Wang, S. (2021b) Unsupervised feature selection via transformed auto-encoder[J]. Knowledge-Based Systems, 215: 106748.
41. Zheng, X., Rong, C, et al. (2019) Foreword to the special issue of green cloud computing: Methodology and practice[J]. Concurrency and Computation: Practice and Experience. 31(23): e5425.
42. Zhou L, Wang L, Ge X, Shi Q (2010) A clustering-Based KNN improved algorithm CLKNN for text classification," 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010), Wuhan, 2010, pp. 212-215, doi: 10.1109/CAR.2010.5456668.
43. Zhou Y, Liu X, Li M (2011) Discussion about Problems and Corresponding Countermeasures in Enterprise Informatization Construction," 2011 International Conference on Business Computing and Global Informatization, Shanghai, 2011, pp. 243-246, doi: 10.1109/BCGIn.2011.69.
44. Zou, W., Guo, L., Huang, P., et al. (2022) Linear time algorithm for computing min-max movement of sink-based mobile sensors for line barrier coverage[J]. Concurrency and

Computation: Practice and Experience, 34(2): e6175.

Dong Chongjie, Master graduated from Guangdong University of Technology in 2010.He is working in Dongguan Polytechnic now. He is an associate professor, and his main research direction is artificial intelligence technology application and database technology.

Zhang Kai received his master's degree from South China University of Technology in 2010. At present, he is a teacher of software technology in Guangdong Communication Polytechnic. His main research interests include artificial intelligence applications, mobile application development, and the application of big data technology.

Received: July 14, 2022; Accepted: December 18, 2022.

Computer Science and Information Systems 20(2):723–741 https://doi.org/10.2298/CSIS220718020Z

Using Deep Learning to Automatic Inspection System of Printed Circuit Board in Manufacturing Industry under the Internet of Things

Kai Zhang

School of Information, Guangdong Communication Polytechnic,
Guangzhou, 510650, China
gdcpzk@qq.com

Abstract. Industry 4.0 is currently the goal of many factories, promoting manufacturing factories and sustainable operation. Automated Optical Inspection (AOI) is a part of automation. Products in the production line are usually inspected visually by operators. Due to human fatigue and inconsistent standards, product inspections still have defects. In this study, the sample component assembly printed circuit board (PCB), PCB provided by the company was tested for surface components. The types of defects on the surface of the PCB include missing parts, multiple parts, and wrong parts. At present, the company is still using visual inspection by operators, the PCB surface components are more complex. In order to reduce labor costs and save the development time required for different printed circuit boards. In the proposed method, we use digital image processing, positioning correction algorithm, and deep learning YOLO for identification, and use 450 images and 10500 components of the PCB samples. The result and contribution of this paper shows the total image recognition rate is 92% and the total component recognition rate reaches 99%, and they are effective. It could use on PCB for different light, different color backplanes, and different material numbers, and the detection compatibility reaches 98%.

Keywords: Deep learning, Digital image processing, Printed Circuit Board (PCB), Automatic inspection system.

1. Introduction

Under the rapid development of industry 4.0, automatic factory began to be implemented in major factories. Robots will replace operators. In the future, robots are more likely to be used in the service industry and even appear at home to help people live a better life. The products produced in the factory will require a lot of manpower to check and remove defective products. The problem scenario is printed circuit board (PCB) proposed in this research has extremely small and complex components. Due to the fatigue of the operator's visual inspection, the quality of the product gate will be affected to a certain extent after a long time of work, even due to people. Different standards are different, automated inspection systems will be used to solve many of these problems, make product inspections more consistent and more efficiently, and can

operate for a long time, which will replace the entire operating system and bring greater benefits [16, 26, 27].

The commonly used equipment for printed circuit board inspection is In-Circuit-Test (ICT) and Function Verification Test (FVT). The functions that the PCB inspection machine can detect are open circuit, short circuit, faulty parts, missing parts, measuring resistance. Measuring capacitance, measuring diodes, and measuring various electronic parts, some of which use Automatic X-ray Inspection (AXI) to penetrate and inspect the internal quality. At present, Automated Optical Inspection (AOI) has been widely used on PCB, such as missing parts, skewed, standing parts, wrong parts, pin warped, short circuit, solder ball detection. Due to the high price of commercially available printed circuit board inspection machines, the use of Automated Optical Inspection (AOI) can reduce the additional purchase cost of the inspection machine, but it still be due to the different products of the PCB, and the algorithms behind it will also be followed. Change the detection compatibility is low, and often cannot be used on the next product. Therefore, this paper will use image processing, positioning point correction and deep learning to develop a detection system for the components on the PCB [1, 11].

At present, PCB surface defects are inspected visually by operators. Fatigue and personal factors will affect the product yield. The purpose of this research is to solve the problem of PCB inspection inconsistency and long-term operator fatigue. The advantages of the proposed method to develop a detection system for parts and faulty parts, and finally test whether the detection system can be used efficiently on other PCB with different light, different background colors, and different material numbers to verify the compatibility of the detection to reduce equipment purchase and the issue of labor costs [31, 34].

In this study, PCB-related products provided by cooperating manufacturers are used as experimental samples, and the field of view (FOV) size of 24mm x 24mm is tested. Since the development of automated testing systems requires the use of many hardware devices, with optical environment architecture, and no ambient light is strong, this experiment uses image processing and deep learning, and the system parameters are learned through experiments. The conditions are as follows: use a 2/3-inch photosensitive chip, a 8 million pixel industrial color camera, and a CCTV lens with a focal length of 45mm and a 5mm extension ring. Use RGBW four-color bowl-shaped light source with a diameter of 250mm. The captured image format is limited to 2048*2048 pixels bitmap images, using universal robot, using conveyor belt with a width of 450mm. The contribution of this paper shows the total image recognition rate is 92% and the total component recognition rate reaches 99%, and they are effective. It could use on PCB for different light, different color backplanes, and different material numbers, and the detection compatibility reaches 98% [33, 39].

This paper's organization includes in section 1. Include paper background, motivation and purposed. Section 2 literature review will learn about digital image processing (DIP) and deep learning applications. The section 3 research method and experimentation. Section 4 is the results and data of the experiment discussion. Section 5 is the conclusion include contribution, future work and limitation.

2. Literature Review

This research mainly focuses on the provided PCB samples after component assembly, detecting surface component defects, and using robotic arms to assist in the work.

2.1. PCB component inspection

PCB testing mainly includes internal testing and appearance testing. Internal testing is mainly based on testing machines, including ICT or FVT and other equipment, the main content is to detect whether the circuit has an open circuit, short circuit, and the measurement of many electronic components. Appearance inspection usually uses AOI, which requires opto-mechanical equipment and image processing algorithms. The main inspection contents are surface scratches, missing parts, excessive tin, and foot deformations. Solder ball inspection can use opto-mechanical equipment to produce suitable solder ball feature images, and detect defects such as tin tip and excessive tin. Recent literature also uses three-dimensional measurement methods to establish its three-dimensional map to detect its complete three-dimensional appearance [4, 12]. At present, there are quite a lot of papers on PCB component detection, most of which are detection using image processing techniques, including image subtraction, component matching [25]. In recent years, the use of machine learning and deep learning has made great contributions to image recognition [9]. Many types of defects have been detected by deep learning. This study intends to use deep learning to detect missing parts, multiple parts, and wrong parts in PCB. Therefore, several articles related to the use of deep learning in PCB component inspection. VGG-16 has the best results, with better accuracy, and can identify up to 25 different components. R-CNN performs positioning, and the mAP is good. RCNN to detect tin on printed circuit boards, the types of defects detected include tin bridges, double tin balls, empty tin. YOLO to detect 9 kinds of capacitors, all of which can be detected. The average time is less than 0.3 seconds. Faster RCNN to identify the components on the PCB and verified that this method is better than template matching. A histogram as the input of a neural network to identify component memory, its method has better success rate. Using neural network to detect whether the solder feet are defective, and finally detect the verification rate is better too.

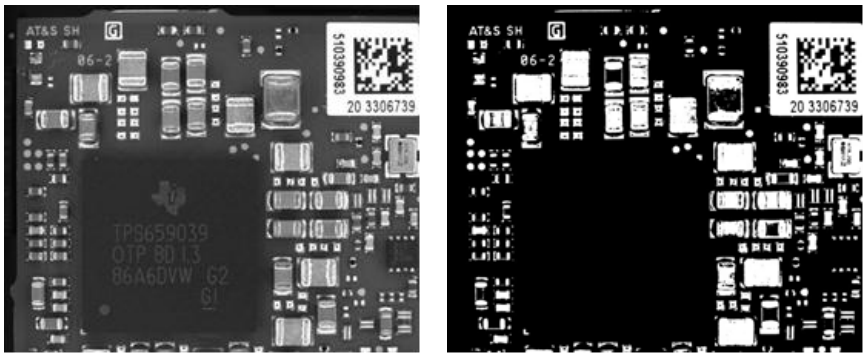
A fast defect detection network used k-means clustering algorithm is used to obtain more reasonable anchors boxes; second, an improved MobileNetV2 is used as the backbone network; after the feature extraction network, the spatial pyramid pooling (SPP) structure is introduced to increase the receptive field of the image [17]. PCB defect detection by using machine learning and other approaches. The current research shows that PCB defect detection using machine learning are miniscule. Early detection is still unexplored and experimented in the industry [40].

2.2. Digital image processing

Digital image processing is a method of image processing. Its concepts include scraping filtering, capturing, segmentation. A publicly available dataset, FICS-PCB, to facilitate

the development of robust methods for PCB-AVI. It has three variable aspects: illumination, image scale, and image sensor [23, 24]. The following will list the calculation logic that will be used in PCB inspection, and refer to Digital Image Processing book content and other literature exploration.

Threshold: Binarization is a common technique in image processing. It can also be said to be a dichotomy. It is often used for grayscale image segmentation to change the color to only pure black and pure white. It can also be used for every color image. Its main purpose is to reduce noise and form a black or white image with strong color contrast to facilitate subsequent morphological processing or feature interception [17-21]. Binarization must be given a threshold. When the input is less than this threshold, its value is 0 (black), otherwise it is 255 (white) as Fig. 1.



(a) Original image (b) Binarized result

Fig. 1. Principle of Binarization (Cheong, 2019)

Contour detection is to find the outer contour of an original image after binarization and morphology. For automatic detection, it can detect the pixel size of the defect and determine whether there is a defect beyond the specified range. Other applications of contour detection can find the outermost contour of the target, and know the degree of contour skew. On the one hand, it can detect whether the PCB component is skewed or offset, and on the other hand, it can obtain its coordinates, which can be repositioned and corrected to be measured. This method is usually used for image subtraction. First, find two points that are more accurate and have little variability, and connect these two points to form a vector, and then use this vector to translate and rotate the image to correct it [5, 13, 37, 38, 45].

Image subtraction is a very practical technique. It is a subtraction of two images taken at different points in the same environment, the same camera and the same light source. This technique is more commonly used in three scenarios. One is to solve the problem of light and shadow. Each image usually has uneven light and shadow. The threshold of binarization is difficult to preset. If you first take a background image, when the image of the object is subtracted from the background image, a more uniform effect can be achieved. The second is to find a moving target. In the case of continuous orientation, this image is subtracted from the previous image to obtain the trajectory of the moving object, and contour detection is used to frame it to obtain the moving target. The third method is used to detect defects [2, 24, 30, 44]. The method is the same as the first

method. The difference is that first take a standard image of the product without defects, and correct the positioning. When the image of the object to be tested is taken, it is compared with the standard image. By subtracting the image, the defect can be obtained, which can be used for defects such as missing parts of the PCB, severely distorted parts, and short circuits on the circuit.

2.3. Deep Learning Applications

With the advancement of neural networks, it can help automated inspections to identify defects more effectively. Back Propagation Network (BPN) [14] and even Convolutional Neural Networks (CNN), CNN is a major breakthrough. Almost all networks are based on CNN to improve and develop. The detection network used in this research, YOLO (You Only Look Once) is also developed based on CNN [29].

Convolutional Neural Networks: In the original machine learning, the image needs to be flattened into one-dimensional information and then calculated, but this method will lose the original image characteristics, and the convolutional neural network Road is identified by using high-dimensional feature information in the image. For example, humans see birds because they see the beak or the back of the chair. The convolutional neural network architecture includes: Convolutional layer: Given one or more filters, extract each feature in the image according to its size, and get feature map. Pooling layer: Retains the important information left after the convolutional layer. Its advantages are reduce parameters to speed up calculations. If there is a slight change between adjacent pixels, it will have little effect on the output result of the pooling layer and reduce overfitting. Fully connected layer: The flatten the remaining features, performs common neural network operations, and classifies them. CNN goes through a multi-layer convolutional layer and a pooling layer. The convolutional layer is responsible for extracting image features, and then leaving important information after the pooling layer [16]. Finally, the fully connected layer performs the final calculation and classification. The proposed convolutional neural network architecture uses high-dimensional feature extraction and has good recognition capabilities.

YOLO: Image recognition has been widely used similar to neural networks, but CNN can only take one image as input and one output at a time. There is no way to recognize multiple items at the same time, while You Only Look Once (YOLO) can recognize multiple objects at once. This study uses YOLOv3. YOLOv1 was proposed by Joseph Redmon [28, 35], and Fig. 2 is the network architecture diagram. Output an image with an input size of 448x448 as SxS grids (grid; the default is 7x7), and then detect whether there is an object in each grid separately, and generate B (default is 2) bounding boxes and N (default is 20) category of conditional class probabilities, the five prediction parameters in bounding boxes are x, y, w, h and confidence scores, x and y are the coordinates of the bounding boxes, w and h are the width and height of the bounding boxes, confidence scores is the value of whether there is an object 0 or 1, conditional class probabilities is the probability of N categories, and finally the bounding boxes with the object are left, and then use Non-Maximum Suppression (NMS) to select the most suitable frame.

The improvement content of YOLOv2 is to add anchor boxes, the output scale is 13x13 and each grid has 5 anchor boxes to predict bounding boxes, so the maximum

output bounding boxes is 845, and the image input size is not limited to 448x448, but a multiple of 32 is the basic structure to change from the original GoogleNet to VGG-16, and Batch Normalization is added to prevent over-fitting and multi-scale training to improve the detection effect. The improvement content of YOLOv3 is to change the output scale to 13x13, 26x26, 52x52, and each grid has 3 anchor boxes to predict bounding boxes. The Feature Pyramid Networks multi-level prediction architecture improves the detection ability of small objects. The maximum number of output bounding boxes is 3549, the loss function is changed from the original sum-squared error to binary cross-entropy, the output activation function is changed from the original softmax to logistic, and the basic architecture is changed from the original VGG-16 to ResNet.

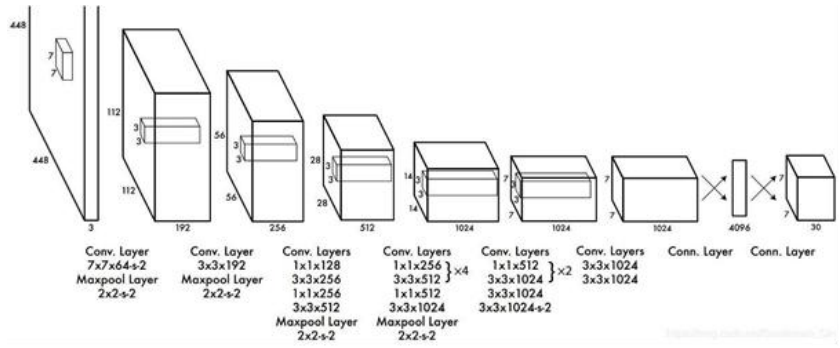


Fig. 2. YOLO network architecture diagram (Ki, 2017)

3. Research Methods

To sorts out the PCB detection methods belonging to the sample, uses robotic arms to work together, constructs a flowchart to describe the steps, to complete the automated detection system, and test whether this system can be effectively used on other PCBs.

3.1. Process of Automated Testing System

The automated inspection system is divided into two steps. The first step is the communication between the image capture operation and the hardware, and the second step is the image post-processing, it is the software inspection. PCB input, the personnel will place several PCBs in the fixture and place the fixture on the conveyor belt. When the sensor knows that the fixture has reached the inspectable range, stop the conveyor belt operation, use the video camera to locate the fixture position, and return it. The coordinates are captured by a robotic arm equipped with an industrial camera and light source. Since the PCB parts are extremely small, this capturing operation will divide a piece of PCB into ten small images.

Fig. 3 is the flow chart of the PCB inspection system. After the image is input, the first operation is performed by the digital image processing (DIP), and then the processed image is input to YOLO, and the components in the image are picked out, and then the components and the original image are transferred to calibration and positioning component algorithm, finally draw the frame on the original image and establish the component coordinates, and determine whether all components have been picked out [6-8, 22, 32, 42]. If one or more components are not picked out, it will be regarded as a defect, and the defective PCB will be rejected. And keep this detection image to facilitate verification whether it is a misjudgment.

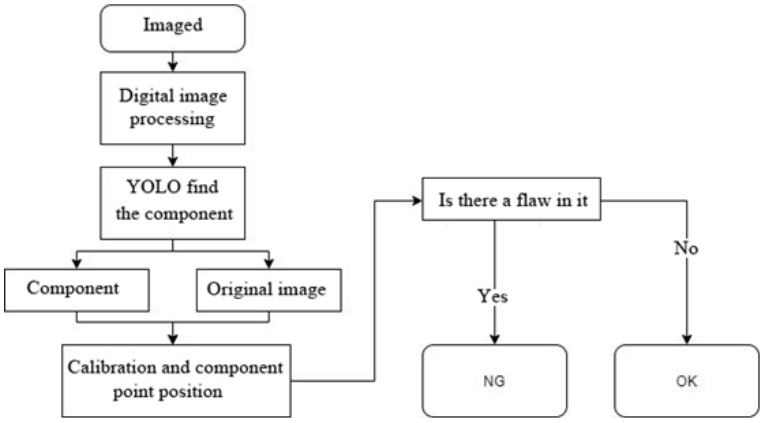


Fig. 3. Flow chart of the PCB inspection system

3.2. Hardware Architecture Framework and Process

The sample PCB size is 59mm x 50mm. The sensor detects the fixture, sends a signal to stop the conveyor belt, and returns a message. The Input/Output is controlled by Arduino. The lower left is Arduino, the upper left component is an infrared sensor, and the lower right component is a relay for circuit switching. When the sensor detects the fixture, the relay turns off the power supply of the conveyor belt. The coordinate position of the entire fixture is located by the video camera above the mechanism and sent back to the robotic arm. The robotic arm moves to the position of each PCB, and then follows the upper left, upper, upper right, left middle, middle, right middle, lower left, bottom, and right take the next image, and finally take another one on the largest chip. Each PCB has a total of 10 images, a fixture has a total of 10 PCBs, and a total of 100 images per fixture. This research has been cross-validated, using industrial camera with 45mm CCTV lens, and with past experience, using HZTEST's AOI-220-RGBW four-angle four-color bowl light source, The characteristics of the solder ball can be effectively shot to facilitate future inspections.

3.3. Detection Architecture

The detection architecture is divided into two phases, the first phase is the model training phase, and the second phase is the detection component phase. The detection component stage is subdivided into three stages. The first stage is image pre-processing, the second stage uses model detection components, and the third stage data post-processing and output. This experiment will build software to implement digital image processing with the python library provided by OpenCV, and use Keras, Tensorflow.

This research uses YOLO to pick out the components. Label each component, use YOLO to train its samples, and predict each component, thereby knowing the location of each component. The model training stage is subdivided into two stages. The first stage is to number the components, and the second stage is to divide the components into a total number of methods and models. In the first stage, the components are numbered. In this experiment, the components on the PCB board are divided into 11 types, which are A, B, C, D, E,...,K according to the size of the components, as shown in Fig. 4.

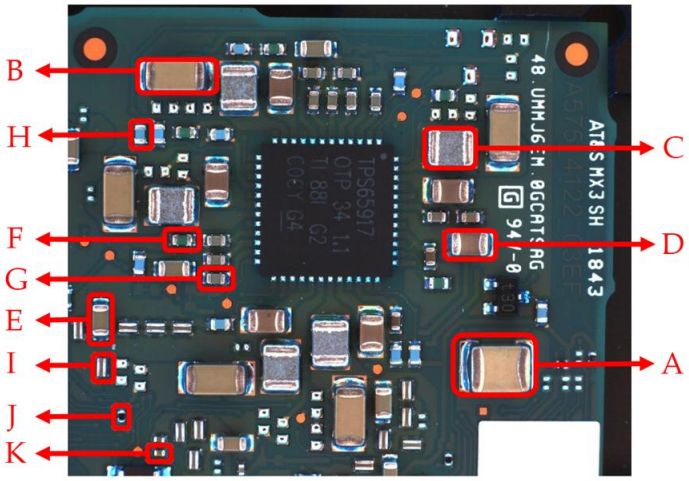


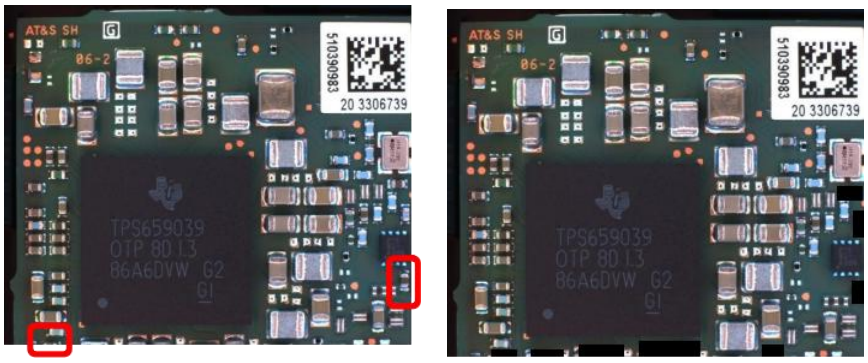
Fig. 4 Classification diagram of component categories

The second stage is to divide the components into several methods and models. The experiment has 180 image samples. Since each image needs to be manually marked with software before training, it is not easy to obtain training samples. There are as many as five components to be marked on each image. There are more than a dozen pieces, 15 pieces of training data and 160 pieces of verification data are taken out randomly. The number of components in these 15 images. YOLO is the object that has been marked for training, and each component can be regarded as a training session. The experiment is divided into Table 2, according to the appearance and size of the components. There are 3 ways, and the group in each way is a detection model.

PCB image pre-processing there are two reasons why the image needs pre-processing. One is that some components of the image may be cut, and the other is that the components are too close to the edge of the image. It may happen that some images

have components and some are absent, as shown in Fig. 5(a). The image is pre-processed to mask the cut components and the components adjacent to the edge, as shown in Fig. 5(b).

Post-processing of data includes calibration and positioning, drawing component outlines, and calculating the number of components. After picking out the component by YOLO, the category information and coordinate position of the component are obtained. Since the captured images will have some errors, each image needs to be corrected, and each PCB has its positioning point, which can be used for correction, but because the PCB has been divided into 9 images, it is no longer available. The method currently used is the end point and start point of the circuit on the PCB, and a two-point vector is used for correction. The corrected image can obtain the coordinate position of the component with low error, and then calculate whether the number of components is correct and mark its outline. If all the components are correct, it is regarded as a good product. If there are missing parts, the image will be picked out and saved for subsequent re-inspection.



(a) Close to the edge and cut element (b) Masked image

Fig. 5. Image pre-processing

4. Experimental Results and Discussion

The experiment is trained by the classification method of model training. It is divided into two stages. The first stage is to test its recognition rate, and the second stage is to test its compatibility. The following will display the experimental results and analyze.

4.1. Parameters Setting and Testing Experiment

YOLO sample and parameter setting, the experiment has 180 image samples, which are divided into 15 as training data and 160 as verification data. The number of training components is 650, the number of anchors is 9, the batch size is 3, and the learning rate is iterative. If the number of times is large, the initial learning rate is 0.001. The number

of iterations stops after 100 iterations after Loss is no longer reduced, and the model weight is stored when loss is lower. YOLO model training results and recognition rate, according to the 165 verification images, if the image contains the specified component of the model, and the component is all detected, there is no missed inspection or multiple inspections, and the number is correct, the identification is deemed correct. Otherwise, the identification is incorrect. The training results and recognition rate of the model are displayed below, and the confidence score of each model is set to 0.5. Table 1 uses one model identification for all components, and there are verification images of missing components among the 160 verification images up to 148 images, as shown in Fig. 6 (a), the correct identification images are all images with fewer components, as shown in Fig. 6 (b), and from the training error of model in Fig.7, it can be seen The loss has decreased, but the convergence is not good, and the recognition rate of this model is low.

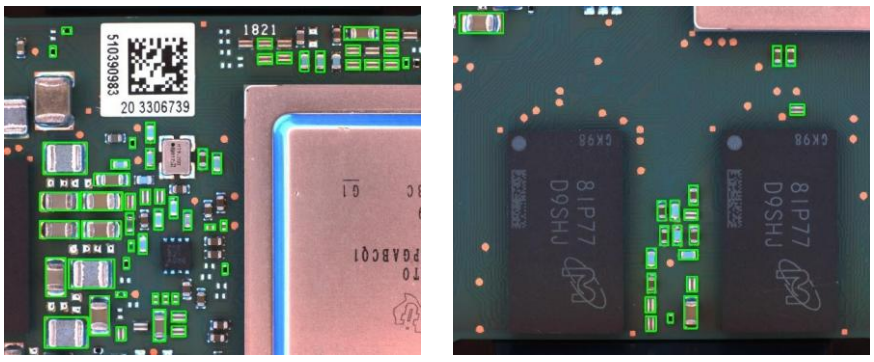
PCB Defect Detection recognition rate experiment: To judge the quality of a PCB detection model is not only to judge the detection effect of a single type of target. Using the original Yolo network respectively, due to the particularity of the defect detection recognition rate is added to the experiment. The evaluation standard is characterized by mean of recognition rate (RR) or identification rate. The higher the value of RR, the better the detection effect and the stronger the comprehensive performance of the PCB for the detection of different types of targets. The RR calculation formula is shown in formula 1.

$$mRR = \frac{1}{N}\sum_{i \in N} RR(i) \tag{1}$$

In the formula, RR represents the average recognition rate, N represents the number of detected target types, and RR is the mean value of different types of RRs. The test results are shown in table 1. The network improved in this paper has better performance in the data level.

Table 1. Model training results of method A1

Model	Identification component	Loss	Identification rate (%)
A1	A,B,C,D,E,F,G,H,I,J,K	46.7917	0.103



(a) Component missed image (b) Unmissed image

Fig. 6. Image recognition result of model No. A1

Table 2 shows the training results of two models with large and small components. The recognition rate of the model number A2-1 has reached 0.909, as shown in Fig. 8(a), 15 images were missed out of the 160 verification images, but the model number A2-2 was identified the rate is only 0.460, and there are many missed detections. As shown in Fig. 8(b), the model A2-1 converges well. It can be seen that the loss decreases more steadily. Higher and unstable decline, this model has a low recognition rate.

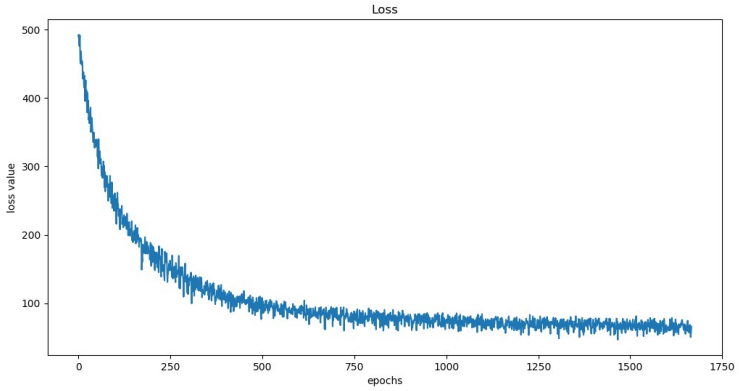
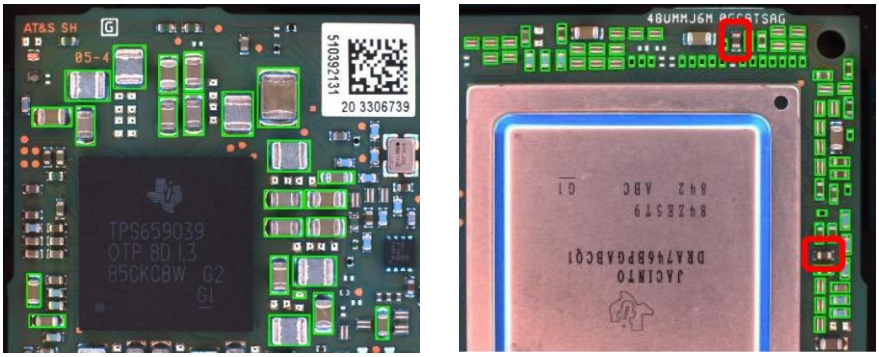


Fig. 7. The training error diagram of the model number A1

Table 2. Model training results of Method A2

Model	Identification component	Loss	Identification rate (%)
A2-1	A,B,C,D,E	13.387	0.909
A2-2	F,G,H,I,J,K	53.5868	0.460



(a) Identification result of model A2-1 (b) Missed image of model A2-2

Fig. 8. Image recognition results of A2-1 model and A2-2 model

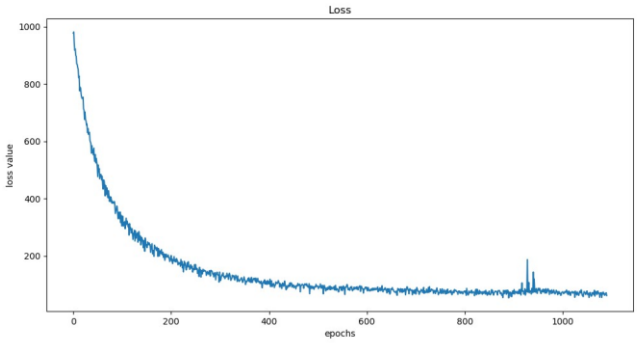
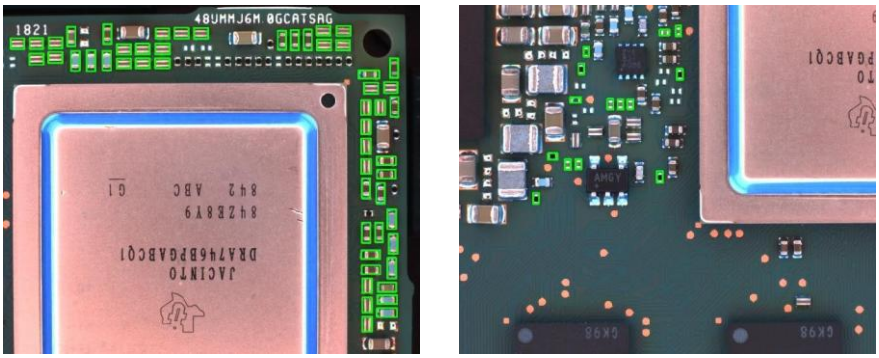


Fig. 9. The training error diagram of the model number A2-2

Table 3 shows the training results of the components divided into three models: large, medium and small, the model number A3-1 is the same as the model number A2-1. The model has a high recognition rate. Compared with the model A2-2 as Fig. 9, the model A3-2 and the A3-3 model have a higher recognition rate. The recognition rate, the model converges well. Fig. 10 shows the identification results of the two models. From the training error graphs in Fig. 11, the loss convergence is relatively stable and low.

Table 3. Model training results of Method A3

Model	Identification component	Loss	Identification rate (%)
A3-1	A,B,C,D,E	13.387	0.92
A3-2	F,G,H,I,	23.9571	0.89
A3-3	J,K	12.3623	0.88



(a) Identification result of model A3-2 (b) Identification result of model A3-3

Fig. 10. The image recognition result of model A3-2 and model A3-3

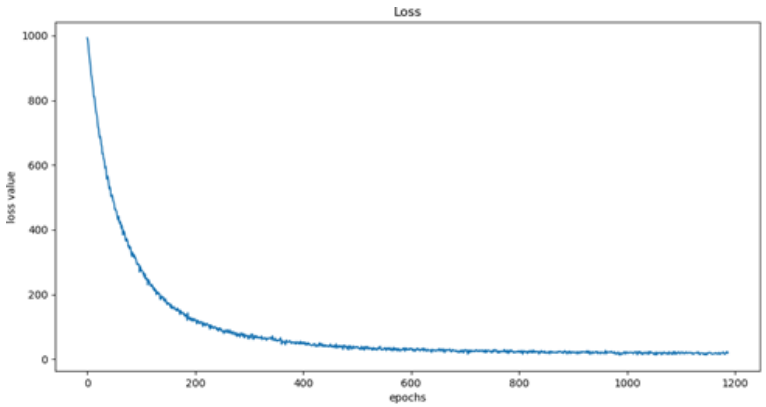


Fig. 11. Training error graph of model A3-3

The three methods proposed for component detection are the method in which all components are identified as one model. A1 component division the method of identifying two models of large and small components. A2 the method of identifying components into three models of large, medium and small. Table 4 shows the detection time and overall identification rate of the three methods. The overall identification rate is to use the full model of this method to identify all components. , If there is no missed inspection or multiple inspections in the verification image, and the number is correct, the identification is considered correct. Otherwise, it is an identification error. In two or three models, there will be a problem of missing verification images. The recognition rate is lower than that of the single-number model, but the recognition rate of method A3 is still higher. Among 160 verification images, 121 images were successfully recognized.

Table 4. Recognition time and recognition rate of each method

Method	Number of models	Recognition time/second (median)	Overall recognition rate (%)
A1	1	0.07	0.20
A2	2	0.14	0.64
A3	3	0.21	0.83

4.2. Discussion

Improve PCB recognition rate: From the experimental process, it can be seen that the three model identification effects of method A3 are better. This method is used to adjust the model parameters to improve its identification rate. For the other two methods, the identification rate is low, and more subjective reasons are proposed. In methods A1 and A2, one model or two models are used for identification. When one model is used for training, it is found that the loss convergence is not good. The possible reason is that the amount of data is not enough, or the similarity between components is too high Caused.

The effect of changing it to two models is improved, and then it is changed to three model detection. The factor that does not continue to use more model identification is that the identification time of each image is not expected to be too long, which affects the detection cycle of the entire detection system (Cycle time).

The parameters are adjusted based on the three models in Mode 3, and the algorithm is not adjusted. Adjust the content as 1. Find more suitable Anchors. 2. Find a more appropriate confidence score. 3. Enlarge the input image to improve the recognition rate. The first model with acceptable convergence (identification A, B, C, D, E) still has its confidence score, and the poorer convergence model (identification F, G, H, I) reduces its confidence score to Improve the detection rate, and the third model (identification J, K) has the problem of component multi-judgment. The confidence score of component multi-judgment is between 0.5 and 0.6. Therefore, the confidence score is increased to avoid component multi-judgment.

Situation to increase the detection rate. Table 5 shows the detection components, confidence scores and detection rates corresponding to the three models.

Table 5. Identification results of each model after fine-tuning the confidence score

Method	Number of models	Confidence score	Recognition rate
A1	A,B,C,D,E	0.5	0.93
A2	F,G,H,I	0.2	0.94
A3	J,K	0.6	0.89
Overall recognition rate		0.92	

In the case of a fixed confidence score, if the input image is enlarged, the final output tensor will also be increased to improve the recognition rate, mainly for the two smaller components numbered J and K, and the verification image will be increased to 450. Fig. 12 shows the changes in the detection time and overall recognition rate after taking measures to increase the image input size.

After the model is adjusted, the input image size is 1024x1216x3, 9 anchors, the recognition time is 0.208 seconds, and the recognition rate has been increased from 0.732 to 0.922. The total number of components is 10500, and the number of missing components is only 30. The total components are recognized, the rate is 0.997. According to the current recognition status, the K component has the lowest recognition rate. For the missing image of the K component, the improvement method is to increase the number of samples to improve its convergence effect, or increase the image input size to increase the output image. However, it still depends on the performance of the computer.

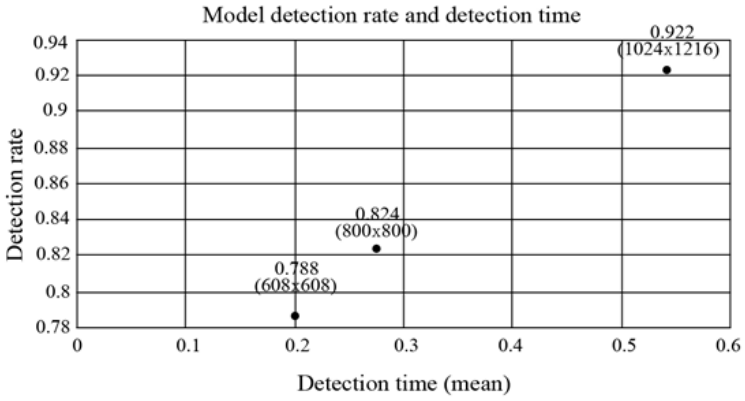


Fig. 12. The change of the recognition rate after fine-tuning the image size

4.3. PCB Testing Inheritance

According to the training results of the model, this study will test whether the model can be effectively used on PCBs with different light, different backplane colors, or other material numbers. The experimental samples have 8 kinds of images with different background, light or material numbers, each of which has 9 images, for a total of 72 images. The recognition rate is 0.903. We use a small number of samples for training, and uses a large number of verification samples to test its recognition rate and detection compatibility. The recognition rate is 0.922 and the compatibility recognition rate is 0.903. If a larger number of training samples are used, and the input image size can be increased, each type can be effectively improved. The confidence score of the component and the overall recognition rate, according to this research, can show that the use of YOLO is an effective compatible model, which can be used on PCB with different light rays, backplanes, and part numbers.

4.4. Comparison of Different Detection Methods

Because the defect of the PCB is the identification target of this paper, the Recognition Rate (RR) of the model is the most important for the identification task of PCB in this paper. The recognition rate can be characterized by RR. The same data set is used in this paper, and R-CNN, SSD, and Yolo are used for PCB detection. The results are shown in Table 6.

The Mean Average Precision (mAP) is our indicator for evaluating model performance our main indicator for evaluate model performance. It is the average of the Average Precision (AP) values of C different defects, and it reflects the accuracy of defect detection as formula 2.

$$mAP = \frac{\sum_{i=1}^{C} AP_i}{C} \qquad (2)$$

Table 6. Defect detection results of different method

Different detection methods	Recognition Rate (RR)%	Mean Average Precision (mAP)
Proposed method	92.2	90.1
SSD	86.9	86.1
R-CNN	91.2	86.2
RetinaNet	88.5	91.1

As can be seen from Table 6, the RR score of Yolo in proposed paper is the highest, reaching 92.2%, followed by R-CNN, reaching 91.2%, indicating that this paper has better detection performance, is faster than other networks. From the perspective of RR, the improved performance is more balanced, it can accelerates the inspection speed and better serve the inspection task of PCB defects, while ensuring the accuracy.

5. Conclusions

Due to the price of ICT machines is too expensive and each inspection line needs to purchase a piece of equipment, and AOI encounters different products and requires different inspection algorithms. PCB surface components are large amounts more complex, and the algorithm is difficult to compile. The research findings deep learning which has achieved considerably in image recognition, has helped a major breakthrough in visual inspection. The contribution of this paper and the results can reduce the fatigue of manual visual inspection and labor costs. Improve detection efficiency and the consistency of testing. Effectively compatible with other PCBs with different lights, backplanes, and material numbers. This detection method can effectively divide components to replace manual visual inspection, and the cost is reduced. It solves the expensive condition of the original ICT machine and is effectively compatible with other PCBs to replace the problem of reprogramming algorithms on AOI. Future work: The current FOV is too small to cover the whole PCB, so it is recommended to increase the camera pixels and improve the computer equipment specifications to facilitate model training. On the PCB, there are not only defects on the components, but still has many surface defects such as open circuits, short circuits, tin tips, scratches, cracks, and foot deformations. It is recommended that the future research direction can use deep learning to detect more surface flaws, such as scratches, foot deformations, and short circuits, and test its recognition rate and compatibility. The limitation is due to the high price of commercially available printed circuit board inspection machines, the use of Automated Optical Inspection (AOI) can reduce the additional purchase cost of the inspection machine, but it still is due to the different products of the PCB, and the algorithms behind it will also be followed. Change the detection compatibility is low, and often cannot be used on the next product.

References

1. Chaudhary V, Dave R, Upla P (2017) Automatic visual inspection of printed circuit board for defect detection and classification, IEEE, pp732 – 737.
2. Cheng, H., Wu, L., Li, R., et al. (2021) Data recovery in wireless sensor networks based on attribute correlation and extremely randomized trees[J]. Journal of Ambient Intelligence and Humanized Computing, 12(1): 245-259.
3. Cheng, Y., Jiang, H., Wang, F., et al. (2019) Using High-Bandwidth Networks Efficiently for Fast Graph Computation[J]. IEEE Transactions on Parallel and Distributed Systems, 30(5): 1170-1183.
4. Cheong K, Suandi A, Rahman S (2019) Defects and Components Recognition in Printed Circuit Boards Using Convolutional Neural Network, LNEE, vol.547: pp75- 81.
5. Dai, Y., Wang, S., Chen, X., et al. (2020) Generative adversarial networks based on Wasserstein distance for knowledge graph embeddings[J]. Knowledge-Based Systems, 190: 105165.
6. Fu, Y. G., Huang, H. Y., Guan. Y., et al. (2021b) EBRB cascade classifier for imbalanced data via rule weight updating[J]. Knowledge-Based Systems, 223: 107010.
7. Fu, Y. G., Ye, J. F., Yin, Z. F., et al. (2021a) Construction of EBRB classifier for imbalanced data based on Fuzzy C-Means clustering[J]. Knowledge-Based Systems, 234: 107590.
8. Fu, Y. G., Zhuang, J. H,, Chen. Y, P., et al. (2020) A framework for optimizing extended belief rule base systems with improved Ball trees[J]. Knowledge-Based Systems, 210: 106484.
9. Gang S, Fabrice N, Chung D, Lee J (2021) Character Recognition of Components Mounted on Printed Circuit Board Using Deep Learning. Sensors 21.9 : 2921.
10. Ki M, Cho H (2017) Recognition of PCB Components Using Faster-RCNN, JKIICE, pp166-169.
11. Li J, Bennett L, Karam L, Pettinato S (2014) Stereo vision based automated solder ball height detection, IPC, Las Vegas, NV, USA.
12. Li T, Guo I (2018) A VGG-16 Based Faster RCNN Model for PCB Error Inspection in Industrial AOI Applications, IEEE
13. Li, X. Y., Lin, W., Liu, X., et al. (2022) Completely Independent Spanning Trees on BCCC Data Center Networks with an Application to Fault-Tolerant Routing[J]. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 33(8): 1939-1952.
14. Lian j, Wang l, Liu t, Ding X, Yu Z (2021) Automatic visual inspection for printed circuit board via novel Mask R-CNN in smart city applications. Sustainable Energy Technologies and Assessments 44 (2021): 101032.
15. Lin B. J., Tsan T. C., Tung T. C., Lee Y. H., Fuh C. S., (2018) Use 3D Convolutional Neural Network to Inspect Solder Ball Defects, ICONIP, pp263-274.
16. Lin Y. L., Chiang Y. M., Hsu H. C., (2018) Capacitor Detection in PCB Using YOLO Algorithm, IEEE, 2018.
17. Liu G, Wen H (2021) Printed circuit board defect detection based on MobileNet-Yolo-Fast. Journal of Electronic Imaging 30.4 (2021): 043004.
18. Liu, G., Chen, X., Zhou, R., et al. (2021) Social learning discrete Particle Swarm Optimization based two-stage X-routing for IC design under Intelligent Edge Computing architecture[J]. Applied Soft Computing. 10, 107215.
19. Liu, G., Chen, Z., Zhuan,g Z., et al. (2020a) A unified algorithm based on HTS and self-adapting PSO for the construction of octagonal and rectilinear SMT[J]. Soft Computing, 24(6): 3943-3961.
20. Liu, G., Zhang, X., Guo, W., et al. (2022a) Timing-Aware Layer Assignment for Advanced Process Technologies Considering Via Pillars. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 41(6): 1957-1970.

21. Liu, G., Zhu, Y., Xu, S., et al. (2022b) PSO-Based Power-Driven X-Routing Algorithm in Semiconductor Design for Predictive Intelligence of IoT Applications. Applied Soft Computing, 114: 108114.
22. Liu, N., Pan, J. Sun, C., et al. (2020c) An efficient surrogate-assisted quasi-affine transformation evolutionary algorithm for expensive optimization problems[J]. Knowledge-Based Systems, 209: 106418.
23. Lu H, Mehta D, Paradis O, Asadizanjani N, Tehranipoor M, Woodard D (2020) FICS-PCB: A Multi-Modal Image Dataset for Automated Printed Circuit Board Visual Inspection. IACR Cryptol. ePrint Arch. 366.
24. Lu, Z., Liu, G., and Wang, S. (2020) Sparse neighbor constrained co-clustering via category consistency learning[J]. Knowledge-Based Systems, 201, 105987.
25. Mogharrebi M, Prabuwono S, Sahran S, Aghamohammadi A (2012) Missing Component Detection on PCB Using Neural Networks, LNEE, vol.134: pp387-394, 2012.
26. Mogharrebi M, Sahran S, Ang C, Prabuwono S (2016) Vision based inspection system for missing footprint detection on printed circuit boards, JATIT, vol.84: pp10- 18.
27. Nti I, Adekoya A, Weyori B, Nyarko-Boateng O (2021) Applications of artificial intelligence in engineering and manufacturing: a systematic review. J Intell Manuf. https://doi.org/10.1007/s10845-021-01771-6
28. Redmon J, Divvala S, Girshick R, Farhadi A, (2016) You Only Look Once: Unified, Real-Time Object Detection, CVPR, Las Vegas, NV, USA, 2016.
29. Richter J, Streitferdt D (2018) Deep Learning Based Fault Correction in 3D Measurements of Printed Circuit Boards, 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2018, pp. 227-232, doi: 10.1109/IEMCON.2018.8614932.
30. Shen, S., Yang, Y., Liu, X. (2021) Toward data privacy preservation with ciphertext update and key rotation for IoT[J]. Concurrency and Computation: Practice and Experience, e6729.
31. Shiina T, Iwahori Y, Kijsirikul B (2018) Defect Classification of Electronic Circuit Board Using Multi-Input Convolutional Neural Network, IJCSE, vol.3, 2018.
32. Song H, Huang G, Chauvel F, Xiong Y, Hu Z, Sun Y, Mei H (2011) Supporting runtime software architecture: A bidirectional-transformation-based approach. Journal of Systems and Software, 84(5): 711-723.
33. Takada Y, Shiina T, Usami H, Iwahori Y, Kamal M (2017) Detection and Classification of Defective Electronic Circuit Boards Using CNN Features and Keypoint Extraction, IARIA, pp113-116.
34. Tsan C, Lin J, Lee H, Tung C, Fuh S, (2018) Solder Ball 3D Reconstruction with X-Ray Images Using Filtered Back Projection, IEEE, pp2505-2510, 2018.
35. Vinita Kaushik S (2016) PCB Fault Detection by Image Subtraction Method, IJSEAS, vol.2, iss.1.
36. Wang C, Jiang B, Lin J, Chu C (2013) Machine vision-based defect detection in IC images using the partial information correlation coefficient", IEEE Trans. Semicond. Manuf., vol. 26, no. 3, pp. 378-384.
37. Wang, S., Wang, Z., Lim, K. L., et al. (2021) Seeded random walk for multi-view semi-supervised classification [J]. Knowledge-Based Systems, 222: 107016.
38. Yu Z, Zheng X, Huang F, et al. (2021) A framework based on sparse representation model for time series prediction in smart city[J]. Frontiers of Computer Science, 15(1): 1-13.
39. Yuk H, Park H, Park S, Baek G (2018) Feature-Learning-Based Printed Circuit Board Inspection via Speeded-Up Robust Features and Random Forest, MDPI, vol.8, iss.6,.
40. Zakaria S, Amir A, Yaakob N, Nazemi S (2019) Automated detection of printed circuit boards defects by using machine learning in electronic manufacturing: Current approaches. Materials Science and Engineering. 767(1), p. 012064. IOP Publishing.
41. Zhang Y, Huang G, Liu X, Zhang W, Mei H, Yang S (2012) Refactoring android Java code for on-demand computation offloading. ACM SIGPLAN Conference on Object-Oriented

Programming, Systems, Languages, and Applications.
42. Zhang, H., Li, J. L,, Liu, X. M., et al. (2021a) Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection[J]. Future Generation Computer Systems, 122: 130-143.
43. Zhang, Y., Lu, Z., and Wang, S. (2021b) Unsupervised feature selection via transformed auto-encoder[J]. Knowledge-Based Systems, 215: 106748.
44. Zheng, X., Rong, C, et al. (2019) Foreword to the special issue of green cloud computing: Methodology and practice[J]. Concurrency and Computation: Practice and Experience. 31(23): e5425.
45. Zou, W., Guo, L., Huang, P., et al. (2022) Linear time algorithm for computing min-max movement of sink-based mobile sensors for line barrier coverage[J]. Concurrency and Computation: Practice and Experience, 34(2): e6175.

Zhang Kai received his master's degree from South China University of Technology in 2010. At present, he is a teacher of software technology in Guangdong Communication Polytechnic. His main research interests include artificial intelligence applications, mobile application development, and the application of big data technology.

Received: July 18, 2022; Accepted: December 22, 2022.

Computer Science and Information Systems 20(2):743–763 https://doi.org/10.2298/CSIS220714015W

Large-scale Image Classification with Multi-perspective Deep Transfer Learning

Bin Wu[1], Tao Zhang[2], and Li Mao[3]

[1] School of Internet of Things Engineering, Jiangnan University,
Wuxi, 214122, China
wubin@jiangnan.edu.cn
[2] China Ship Scientific Research Center,
Wuxi 214122, China
taozhang@jiangnan.edu.cn
[3] School of Artificial Intelligence and Computer Science, Jiangnan University,
Wuxi, 214122, China
wxmaoli@jiangnan.edu.cn

Abstract. Most research efforts on image classification so far have been focused on medium-scale datasets. In addition, there exist other problems, such as difficulty in feature extraction and small sample size. In order to address above difficulties, this paper proposes a multi-perspective convolutional neural network model, which contains channel attention module and spatial attention module. The proposed modules derive attention graphs from channel dimension and spatial dimension respectively, then the input features are selectively learned according to the importance of the features. We explain how the gain in storage can be traded against a loss in accuracy and/or an increase in CPU cost. In addition, we give the interpretability of the model at multiple scales. Quantitative and qualitative experimental results demonstrate that the accuracy of our proposed model can be improved by up to 3.8% and outperforms the state-of-the-art methods.

Keywords: large-scale image classification, channel attention module, spatial attention module, interpretability of the model, multiple scales.

1. Introduction

In computer vision, image classification is a hot subject, and it is also an important foundation in some fields such as object detection [1, 2], face recognition [3], pose estimation [4, 5], population density estimation [6, 7], image segmentation [11, 12] etc. However, due to the large number of image categories and the limitation of computing resources, it is difficult for traditional classification algorithms to achieve a high accuracy. At present, the image classification algorithm mainly relies on the deep neural network model [8, 9, 10, 11]. Different from the traditional image classification method, deep learning scheme does not require complex feature decomposition of the target image. However, the unexplained black box in AI system makes users can only get detection results, and cannot give the reasons [14, 15]. In this regard, the Explainable AI (XAI) is widely developed in recent years [16, 17]. Generally, different decision-makers

focus on different goals in XAI to ensure the high performance and provide a reasonable explainable model synchronously, our work focuses on the robust construction of large-scale image classification model, we also focus on the interpretability of the model at multiple scales.

One main difficult task of image classification is the feature extraction [18, 19]. The so-called feature extraction refers to constructing an algorithm to extract features in the target image, such as the edge feature, the color feature, etc. At present, the best approach is to use deep neural network model to conduct image classification tasks [20, 21]. The experimental result of VGGnet shows that better classification accuracy could be obtained by constructing deeper convolutional neural network model. Inspired by this idea, the deep residual network is constructed by cross-layer connection method, and it achieves higher classification accuracy. GoogLeNet increases the adaptability of the network to different scales, which also obtain better classification accuracy. ResNeXt and Xception added cardinality to the network model, proving that the cardinality can not only reduce the overall parameters of the model, but also has a strong ability of representation. However, considering the fact that the characteristics of different scales are characterized by different feature and highly complementary, it is necessary to consider the complementarity between different feature layers.

The attention mechanism is similar to that of human vision. It depicts a phenomenon, we always focus our attention on the main things to get key information [22]. Attention mechanics have been used in a wide range of scenarios. The neural network could captures more critical information with the help of the attention mechanism. The attention mechanism enables the neural network model to be self-explanatory in the channel dimension and the spatial dimension. In order to give the interpretability of attention mechanism more clearly, the channel attention mechanism (CAM) and the spatial attention mechanism (SAM) are firstly considered in the attention domain. Our attention domain mainly consists of three types: spatial domain, channel domain and mixed domain. In our experiments, it is found that better performance is obtained with using the CAM.

In recent years, many researchers have tried to solve these problems. Wolpert believe that the reason for the evolution of the brain is not for thinking and feeling, but for controlling motion, which is the core idea of the Deep Reinforcement Learning [23, 24]. Curriculum learning and self-paced learning represent the recently proposed learning strategy. Their core idea is to simulate the cognitive mechanism of human beings, they first learn simple and general knowledge structure, then gradually increase the difficulty degree and transition to more complex and professional knowledge. These two methods have great advantages in handling small samples of data and the interpretability of models. However, no one method can solve the problems of small sample size and unrestricted scenes perfectly. In order to clear the aim of this paper, we select some representative works, and provide a comparative analysis of related works in form of table discussing the advantages, disadvantages, techniques, objectives, and limitations etc, as is shown in Table 1.

In view of the existing problems of the field of large scale image classification in complex environments, we construct a novel feature self-selection scheme in the feature layer to get more useful semantic information of the deep network, we try to combine the image loss function and a novel adversarial loss term to optimize our network.

Table 1. Comparative analysis of related works

Algorithms	Advantages	Disadvantages	Techniques	limitations
Detection-based approaches [3-9]	Applicable to simple monitoring environments	The recognition error is high when the background is complex	General feature extraction strategy	Lack of interpretability of constructed model
Attention-based approaches [10-18]	Real-time performance	Work not well under variety of scenes and camera angles	channel attention mechanism, spatial attention mechanism, etc	Complex and unrestricted scenes
Deep learning scheme [19-26]	The detection rate and recognition rate are high and stable	Easy to fall into local optimum, and the decision is short of interpretability	CNN network, RNN network, multi-view deep learning, etc	Lack of interpretability of constructed model
Multi-task learning scheme [28-34]	The model is more stable and can detect multi-scale objects	Training process is complicated and time-consuming, easy to fall into local optimization	learning multiple task classifiers jointly	Complex and unrestricted scenes
Explainable AI [27,35-38]	The interpretable ways of various deep models are given	Theoretical analysis is not enough, Small sample size	Image processing and visual technologies	Complex and unrestricted scenes
Ours	**It combines the advantages of existing algorithms and is suitable for various scenes**	**The real-time performance of the algorithm needs to be improved**	**CNN, Transfer learning, self-paced learning, etc**	**Real-time requirement**

In view of the existing problems of the field of large scale image classification in complex environments, we construct a novel feature self-selection scheme in the feature layer to get more useful semantic information of the deep network, we try to combine the image loss function and a novel adversarial loss term to optimize our network.

Main contributions of this paper are described as follows:

First, our work focuses on the robust construction of image classification model. We propose an multi-channel convolutional neural network model consisting of channel attention module and spatial attention module, which could potentially explain the classification results;

Second, considering the fact that the characteristics of different scales are characterized by different feature and highly complementary, we construct a novel feature self-selection scheme in the feature layer to get more useful semantic information of the deep network;

Third, a nonlinear model based on deep transfer learning is proposed to solve the the classification problem of multiple perspectives, and we construct a novel adversarial loss term to optimize our network;

Last, in order to give the interpretability of the model at multiple scales and reduce the error loss in the constructed model, self-paced learning and transferring learning modules are designed to optimize the constructed model.

In this paper, the traditional deep feature learning is further exploited from the explainable view, and the classification effect is improved by our constructed multi-channel convolutional neural network model consisting of channel attention module and spatial attention module. The main structure of this paper is as follows: Section 2 starts with a brief presentation of necessary related concepts. Section 3 gives our proposed channel and spatial attention module based on explainable machine learning. Designed experiments and discussion are depicted in Section 4. Finally, some conclusions are presented in Section 5.

2. Related Work

Explainable AI (XAI) refers to those Artificial Intelligence techniques aimed at explaining, to a given audience, the details or reasons by which a model produces its output [25]. Consequently, XAI borrows concepts from philosophy, cognitive sciences and social psychology to yield a spectrum of methodological approaches that can provide explainable decisions for users without a strong background on Artificial Intelligence. Therefore, XAI targets at bridging the gap between the complexity of the model to be explained, and the cognitive skills of the audience for which explainability is sought. Interdisciplinary XAI methods have so far embraced assorted elements from multiple disciplines, including signal processing, adversarial learning, visual analytics or cognitive modeling. Although reported XAI advances have risen sharply in recent times [34, 35], there is global consensus around the need for further studies around the explainability of machine learning models. This includes interpretable reasoning of models, neurosymbolic reasoning or systems based on fuzzy rules, etc..

At present, explanations provided by different algorithms are fragmented and independent, which makes it difficult to determine reasonable decisions and explain model structures. Generally, current interpretable classifiers could be classified into the following four kinds: the selection of optimal training set, correlation selection of heat graph, semantic analysis, model visual interpretation [27, 32, 33]. Existing interpretable artifical intelligence models based on the selection of optimal training set can provide the basis behind the classification [34, 37]. However, there is no mechanism for identifying potential misclassification of classifiers in the existing classification model. Warning users about misclassification will help prevent errors from entering the system. One of the reasons for misclassification is the reduction of distance between classes. Some outliers or edge elements of a class can share the common characteristics of adjacent classes. However, there is no mechanism to ensure the number of subclasses of a given class and whether it makes sense to merge closely related subclasses of two adjacent classes into a new class and implement the correct classification. The interpretable artifical intelligence models based on correlation selection of heat graph are another way to understand it. Xu [26] proposes a solution based on database transaction model interpretation, whose explanation is on the basic of logical structure or reasoning. The static structure makes it unsuitable for the deep network classifier. In the constructed model, the system dynamically give appropriate explanations from stored vocabularies, which in turn are generated based on model learning. It provides a consistent view of models and interpretations beyond the scope of existing technology.

Simonyan [29] proposes an interpretable model in which they are interactively validated through visual features and similarity. Moreover, k-means clustering was used to analyze the similar features, so that the average features obtained had greater robustness and relatively low time complexity. In addition, it does not consider the importance or relevance of model, nor does it cluster with respect to output classes.

Some researchers try to use semantic analysis to explain the models. He [28] conducts the explanatory demonstration of the model. By observing the change of the connection mode of the network, the hidden layer is explained visually. However, the feature learned by the network layer are not described in detail. Sengupta [30] proposes an image interpretation and generation method based on semantic analysis. Take an image signature with a fixed length of 8000 to generate a caption. In this model, the correlation of features is firstly determined and the signature generation is carried out on this basis. Since eigenvalues could be of any length, strategies that follow highly correlated features are interpretable. As the power of interpretation becomes more important in intelligent decision-making, AI systems are no longer there to serve as black boxes [31, 37].

Model visual interpretation is also another way to understand the interpretability of the model. Russakovsky [36] proposes a novel model based on the attention mechanism, and it is named the residual attention network. As the network layer deepens, attention modules can extract key information from different layers. Finally, it got 4.8% Top-5 error rate on ImageNet. Hu [38] proposes the SENet 44 network model. In the training process, the model can distinguish the importance of different channels, then enhances useful features and inhibits useless features according to the importance of feature. Finally, it won the ILSVRC2017 classification task championship with a 2.25% Top-5 error rate. Moreover, the text interpretation can't match the characteristics of a certain layer of deep learning network, and it lacks of continuous interpretibility. In general, the text explanation generated for classification comes from training data based on model annotations. Up to now, data labels are set manually and are very subjective, and it doesn't take into account the differences between the different elements. Therefore, it is not possible to determine the relevant region of the image that is most useful for classification. In most cases, experts are encouraged to use their attribute label data as interpretable evidence.

Therefore, this paper proposes a novel image classification framework based on multi-perspective deep transfer learning. Attention mechanism is used to describe varied image characteristics, the self-paced learning strategy is adopted to solve the problems of small number of labeled samples and model interpretation. A nonlinear model based on deep transfer learning is proposed to solve the classification problem of multiple perspectives, also the interpretability of the model is further improved.

3. The Proposed Channel and Spatial Attention Model

In view of the existing problems in the field of image classification in complex environments, this paper considers the solutions based on deep network, and further studies popular algorithms such as deep reinforcement learning, self-paced learning and transfer learning, aiming at the urgent model explanation problems in the framework of

deep network. The overall technical framework containing CAM and SAM is shown in Fig. 1.

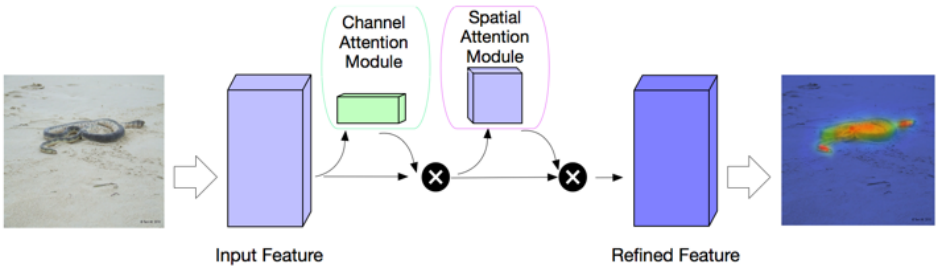


Fig. 1. The overview of model

For the convolutional neural network model, depth, width and attention mechanism are the main factors affecting the accuracy of image classification. At present, attention mechanism contains CAM and SAM. CAM acts on the channel domain, weighting different channel features. For a $C \times H \times W$ feature graph, the weight of channel attention C is different, while the weight of H $H \times W$ is the same. For CAM, the weight of each C on different channel dimensions needs to be learned. To reduce the amount of computation and improve classification accuracy, the pooling layer in the general convolutional neural network directly uses the maximum pooling method or average pooling method to compress the image information. For SAM, only the key information in the spatial features is extracted. We first introduce the general framework of proposed model with CAM and SAM in this section. Finally, we describe how to combine them together.

After convolution operations, an intermediate feature map $F_{\text{in}} \in R^{C \times H \times W}$ is obtained. F_{in} is the input of model. One-dimensional channel map $M_{\text{c}} \in R^{C \times 1 \times 1}$ is inferred after the channel attention module and two-dimensional spatial map $M_{\text{s}} \in R^{1 \times (H \times W) \times (H \times W)}$ is computed, and the corresponding spatial attention module is shown in Fig. 1. The entire attention calculation process could be summarized as:

$$F_1 = M_{\text{c}}(F_{\text{in}}) \otimes F_{\text{in}} \quad (1)$$

$$F_{\text{out}} = M_{\text{s}}(F_1) \otimes F_1 \quad (2)$$

where $\otimes$ denotes element-wise multiplication. F_1 is achieved after F_{in} passing the channel attention module. F_{out} is the final output and the attention value is broadcasted during multiplication process. Attention feature are the multiplication of element levels, which will be propagated automatically, indicating that channel attention broadcasts along spatial dimension, while spatial attention broadcasts along channel dimension.

3.1. Channel Attention Module

The CAM captures the relationship between channel features. It focuses on the key channel information and weakens the influence of the useless channel information [39, 40]. It uses an attention mechanism (similar to the self-attention mechanism, such as query, key, value) to get the similarity between channel graphs, and then use the weight of channel graphs to update. Finally, the matrix of computation attention is obtained, which can enhance the key features. The CAM makes the neural network model pay more attention to the channel features with the key information. On the basis of convolution, we first extrude the feature graph to obtain the global feature of each channel. Then, we use the global feature to get the relationship between different channels and the weight of different channels. Finally, we multiply the weights to get the features on the basis of the original feature graph.

In a convolutional neural network, the convolution operation only performs on image feature space, it is difficult for the convolution module to get the relationship between different feature channels. To get an eigenmatrix, an image needs to go through several convolutional layers and the number of channels represents the number of cores in the convolutional layers. In a normal neural network, the number of convolution kernels is usually as high as 1024 or 2048. Therefore, not each channel is useful for feature extraction. The CAM will help the neural network model select more informative channels. Besides, We encode spatial features using a global average pool that provides feedback for each pixel on the feature map. The following formula shows the global average pool calculation process.

$$F_{\text{avg}} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_{\text{in}}(i.j) \tag{3}$$

where $F_{\text{avg}} \in R^{C \times 1 \times 1}$ represents the result after implementing the global average pooling on the input feature map F_{in}. In order to capture the relationships between different channels, two conditions need to be met for CAM: firstly, it must be flexible, because it needs to learn the nonlinear relationship between different channels; Secondly, the learning relationship is not mutually exclusive, because it allows for a multichannel feature instead of a hot spot form. We describe the channel attention map M_{c} as follows.

$$M_{\text{c}} = \sigma(W_2 \operatorname{Re} LU(W_1 F_{avg})) \tag{4}$$

where $W_1 \in R^{\frac{C}{\text{r}} \times C}$ and $W_2 \in R^{C \times \frac{C}{\text{r}}}$, σ denotes Sigmoid function. W_1 and W_2 are fully connected layers. To improve the explainable ability of the model, we construct two fully connected layers, namely the bottleneck structure. W_1 denotes the dimensionality reduction layer and the dimensionality reduction factor r is a super parameter. Then the Relu function is used and the W_2 layer restores the dimension to the original number. The process is shown in Fig. 2.

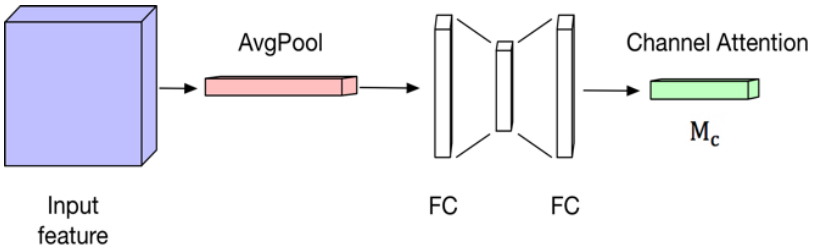


Fig. 2. The details of channel attention module

3.2. Spatial Attention Module

Unlike CAM, SAM only plays the role of distinguishing key information within a single image feature map. First of all, we use average pooling and max pooling to compress the input feature and then we use mean and max operations on the input feature at channel dimension. Finally, we could get two two-dimensional features. Considering different channel size, the two two-dimensional features are combined together to obtain the feature with channel number of 2. And they are convolved to ensure that the resulting features are consistent with the input feature in spatial dimension.

Spatial attention map is generated by adding the internal relationship, as shown in Fig. 3. $A \in R^{C\times H\times W}$ represents the input of the SAM. After conducting convolutional layers, feature maps B, C and D ($(B,C,D) \in R^{C\times H\times W}$) are generated. We reshape B and C to $R^{C\times H\times W}$ and $H\times W$ represents the resolution in spatial module.

Then we get $R^{(H\times W)\times(H\times W)}$ by operating a matrix multiplication between the transpose of B and C. The spatial attention map M_c could be obtained when a softmax layer is applied. M_c is computed as follows:

$$M_{sij} = \frac{\exp(B_i \times C_j)}{\sum_{i=1}^{H\times W} \exp(B_i \times C_j)} \tag{5}$$

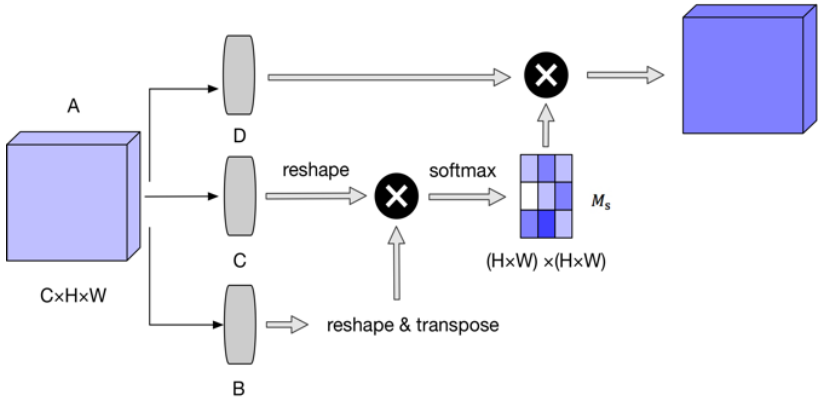


Fig. 3. The details of spatial attention module

3.3. Explainable Object Detection Model of Combing Self-paced Learning and Deep Reinforcement Learning

The deep network model is designed for learning across model characteristics. This paper proposes a new application based on the deep network framework, using deep network to learn multi-mode. In particular, this paper demonstrates that cross-modal feature learning could effectively relate different features to obtain a discriminant feature. In addition, the paper designed how to learn a shared feature between multiple modes and evaluate it on a particular task. Multi-mode explainable deep network model can be seen in Fig. 4. This model consists of three streams, video information, text information and audio information. The structure of the three streams is identical, each consisting of eight layers (including the input layer).

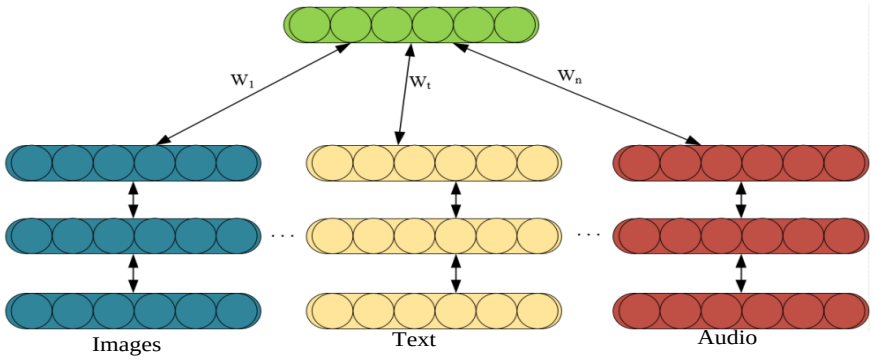


Fig. 4. Proposed multi-mode explainable deep network model

There are two problems with the traditional multimodal model. First, there is no clear goal for the model to find correlations across modes. Some hidden layer units adjust the parameters only for one stream, and others adjust the parameters only for other stream, so that it is possible for the model to find the desired feature. Second, there is only one mode for supervised training and testing in the cross-modal learning arrangement, which makes the model unexplainable. If there is only one modal representation, it is necessary to integrate observable variables that are not observed. Therefore, this paper proposes a deep self-coding model to solve the above problems. Inspired by the noise-reducing self-coding model [41], this paper proposes a training multi-mode deep self-coding model (Fig. 4), which uses an extended (extended single-mode input) but noisy data set. In fact, the model is still required to reconstruct the three modes when one mode uses zero as input and the other uses the original value as input when expanding. Therefore, one-third of the training data is input only by video, one-third of the training data is input only by text, and the last third is input only by voice. This model can be viewed as an example of multitasking learning.

Algorithm 1: Training process of our proposed network

Input: training data set $D\{(x_1,l_1),(x_2,l_2),\Lambda,(x_t,l_T)\}$

Weight K

Initialize the experience pool *M*

Randomly initialize network parameters θ

for k=1 to K do :

Scramble training data sets D

Initialize the initial state s_1 to x_1

for t=1 to T do :

Choose an action based on the ε-greed strategy

$a_t = \pi_\theta(s_t)$

$r_t, \text{terminal}_t = \text{Reward}(a_t, l_t)$

Set the state s_{t+1} to x_{t+1}

Store experience data into the experience pool M

from M

Set y_j as:

If $\text{terminal} = \text{True}$, set as r_j ;

If $\text{terminal} = \text{False}$, set as $r_j + \gamma \max_a Q(s, a_j, \theta_{k-1})$

Gradient descent

$L_\theta = (y_i - Q(s_j, a_j; \theta_{k-1}))^2$

If $\text{terminal}_t = \text{True}$,

break

When designing the intensification strategy, this paper uses the Q network to interact with its environment during the data generation phase. The system looks at the current scene, which consists of multi-mode data, and takes actions using the ε-greedy strategy. This environment in turn provides scalar rewards. Interaction experiences are stored in replay memory M. Replaying M preserves N recent experiences, which are then used to update the network parameters during the training phase. In the training stage, the network structure will use the data stored in replay memory M to train the network. Assume that the superparameter n represents the number of experiences replay, and for

each experience replay, a mini-cache B containing several interactions is randomly sampled from the finite size replay memory M. The model will be trained by sampling from cache B, and the parameters of the network will be updated iteratively in the direction of The Behrman target. The algorithm is divided into two phases to avoid latency. Therefore, this paper divides the algorithm into two stages: in the first stage, the robot collects data through limited time interaction with human beings; In the second stage, the training phase is activated to train the multimodal depth Q network.

In order to avoid non-convex optimization problems falling into poor local solutions, the proposed network optimization method adopts multiple random initializations to train the model, and then chooses the initialization network with the best performance to construct the model. However, this method is too adhoc and the calculation cost is too high. Self-learning is the best solution to non-convex optimization problems. The aim of curriculum learning is to simulate the cognitive mechanism of human beings by first learning simple and universal knowledge structure and then gradually increasing the difficulty to learn more complex and specialized knowledge. However, self-paced learning has been improved in curriculum learning. Instead of assigning prior knowledge to sample learning sequence in advance, the learning algorithm itself determines the next learning sample in each iteration. Detailed algorithm is described as Algorithm 1.

As mentioned in the introduction part, the traditional method uses L2-based regression to train a multi-scale network and finally forms a fused prediction density map. It also mentions why it is only based on L2 regression. In order to solve such problems and make the final density clearer, we choose to use adversarial loss. The adversarial generative network involves the model generator G and model discriminator D. The two are like playing a minmax game: the image generated by the training generator G deceives model D, while the aim of training D is to distinguish the synthesized image from the actual image, if it is inconsistent, it is Fake, and if it is consistent, it is True [42]. In our method, this adversarial loss is defined as follows:

$$L_A(G,D) = E_{x,y\sim P_{data}(x,y)}[LogD(x,y)] + E_{x\sim P_{data}(x)}[Log(1-D(x,G(x)))]. \quad (6)$$

Where x denotes the training patch and y denotes the corresponding ground heat map. The aim of G is to minimize this goal, and D tries to maximize it.

The traditional pixel-by-pixel Euclidean loss is based on large deviations between pixels, it will make the feature map fuzzy when facing sharp edges or outliers, and thus the generated density map will become fuzzy. But the adversarial loss discards the large deviation between the existing pixels. It is a binary judgment for each pixel, either true or false.

Because of the lack of punishment constraint on the ground real image, only using adversarial losses can sometimes lead to abnormal spatial structure. As suggested in previous work, we also use two common losses to smooth the solution. Details are as follows:

• Euclidean loss: In our model, the L2 loss is also adopted to change the estimated density map G into the discriminator model D to approximate the basic facts in the sense of L2. Assuming that W×H resolution image with c channels is constant, we design the following rule to depict the pixel-by-pixel loss:

$$L_E(G) = \frac{1}{C}\sum_{c=1}^{c} \| P^G(c) - P^{GT}(c) \|_2^2 . \quad (7)$$

Where $P^G(c)$ represents the pixel generating the heat map and $P^{GT}(c)$ indicates the pixel of the ground truth map, here we set C = 3.

• Perceptual loss: This kind of loss function was originally added into the image task by Johnson [43]. For image conversion and super-resolution tasks, it compares the features obtained by convolution of the real picture with the features obtained by the convolution of the generated picture, making the high-level information (content and global structure) more similar, whose aim is to minimize the perceived difference between the two. The definition of perceived loss is as follows:

$$L_p(G) = \frac{1}{c}\sum_{c=1}^{c} \| f^G(c) - f^{GT}(c) \|_2^2 . \quad (8)$$

Where $f^G(c)$ represents the pixels in the advanced receptiveness features of the previous heat map and $f^{GT}(c)$ denotes the pixels in the following receptiveness features of the ground truth, it should be noted that C is set to 128.

So the overall first-stage loss could be defined as follows:

$$L_I = ArgMin_G Max_D L_A(G,D) + \lambda_e L_E(G) + \lambda_p L_p(G). \quad (9)$$

Among them, λ_e and λ_p are the weights that measure Euclidean loss and perceptual loss. In this experiment, we set $\lambda_e = \lambda_p = 150$.

During our model training process, the patches are sent to G-large and G-small, respectively. To obtain the estimated heat map P-parent and generated by G-small at the end of the subnetwork, the four sub-pictures P-child are spliced to form P-concat. The constraint of cross-scale consistency is imposed on P-concat and P-parent. In general, given the W×H density map with c channels, the cross-scale consistency constraint based on L2 loss is defined as follows:

$$L_c(G) = \frac{1}{c}\sum_{c=1}^{c} \| P^{prt}(c) - P^{cnt}(c) \|_2^2 . \quad (10)$$

Where $P^{prt}(c)$ denotes the pixels in the parent block heat map, and $P^{cnt}(c)$ denotes corresponding pixels after the child block density map is spliced, C is set to 3.

Algorithm 2: Reward scheme

Deifne Reward function $(a_t \in A, l_t \in L)$:
  Initialize Terminal as False
  If $s_t \in D_p$,then
    If $a_t = l_t$, then
      Set r_t as 1
    else
      Set r_t as -1
      $terminal_t = True$,
  else
    if $a_t = l_t$, then
      Set r_t as λ
    else
      Set r_t as $-\lambda$
  return r_t , $terminal_t$

The final goal is to combine the four loss functions mentioned above to achieve the final loss function:

$$L_{II} = L_1 + \lambda_c L_c(G). \quad (11)$$

Among them, λ_c is a predefined weight to achieve cross-scale consistency with respect to constructed loss function. It's worth noting that if λ_c is set to 0, the two normal forms will be generated independently. In addition, we define the corresponding reward scheme, as is shown in Algorithm 2.

4. Experimental Results and Analysis

4.1. Datasets and Experiments Settings

The dataset is used in the experiment including ImageNet- 1K and Cifar-10050 [29, 31, 44, 45, 46]. The Cifar-100 contains 100 classes and each class has 600 color images. But each images is only a size of 32*32. Five hundred images in each class serve as the training set and the rest as test set. For each image, it has two labels, fine-labels and coarse-labels, which represent the fine-grained and coarse-grained labels of the image respectively, and Cifar-100 is hierarchical. In Fig. 5, we extracted the images from Cifar-100 as the visual example.

ImageNet-1K is an image dataset and each concept image is quality-controlled and manually tagged. At present, ImageNet-1K consists of 1,4197,122 images. The major categories include: animal, bird, fish, flower etc. In Fig. 6 shows the visual examples of the ImageNet-1K.

In experiment, we compare the effectiveness between the CAM and the SAM. We compared 4 different network models: baseline network, baseline network with CAM, baseline network with SAM, baseline network with CSM. For the unbalanced data classification model based on our proposed model, we uses the ε-greedy strategy to determine the parameters of models. The probability of exploration decays linearly from 1.0 to 0.01. The size of the experience pool is set to 50,000, and the interaction between intelligent agent and the environment is about 120,000 steps. The discount factor for instant rewards is set to 0.1. The gradient of cost function to parameters is calculated by using the data of the whole training set. Adadelta algorithm [25] is used to update network parameters (aimed at solving the problem of plummeting learning rates), and the learning rate is 0.002. For other algorithms, the optimizer is Adam [44], the learning rate is 0.0005, and the batch size is 64. Training rounds are 100.

In this comparative experiment with the state the art, resNET-50 and RESNET-200 models were used to validate this strategy on CIFAR-10 and ImageNet-1Kdataset. Epoch was set to 270, batch size was set to 4096, and learning rate was set to 0.01. The comparison was done based on the same datasets and rules.

Finally, we make some comparisons between our method and existing methods, and we draw the following conclusions:

1. Data enhancement is relatively complete, and the scale variation of multi-scale training is relatively small;

2, the learning rate is directly default, without warm up and decay;

3. The weighted processing of loss is similar in the four datasets, and some are not even set;

4. Anchor matching strategy is also similar after verification, for example. each YOLO layer matches the largest IOU.

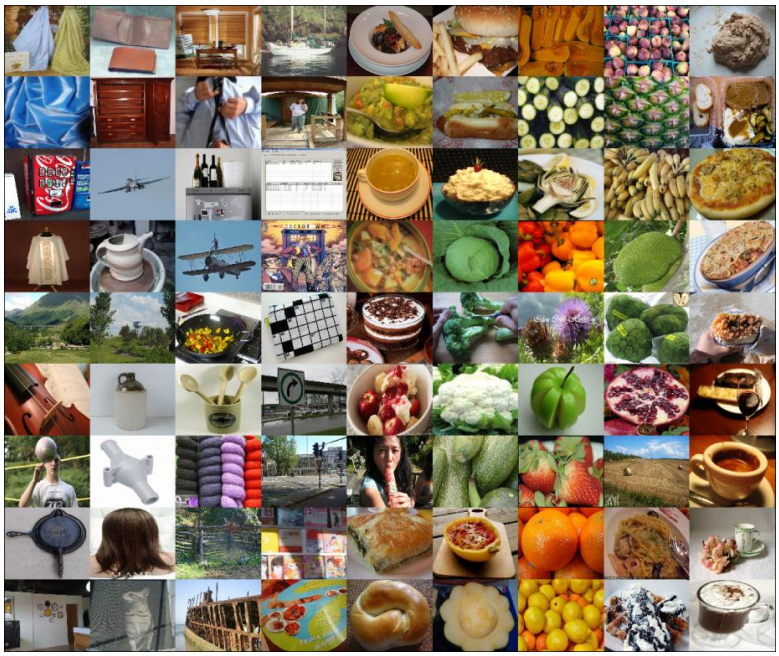


Fig. 5. Visual Examples of the Cifar-100 [31]

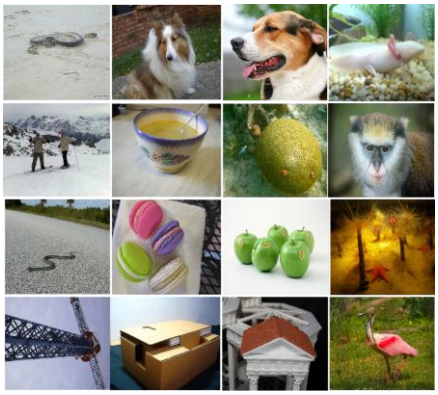


Fig. 6. Visual examples of the ImageNet-1K

4.2. Image Classification on ImageNet-1K

In this part, we use ResNet and WideResNet as the baseline model. On this basis, we add attention mechanism for comparison. The extensive image classification experiments are conducted based on the ImageNet-1K. The structure of ResNet with adding SE module is shown in Figure 7 and the results of the experiment is shown in Table **2**. The experiment still prove that networks with CSM performs better than the baseline module, indicating that attention mechanism can be well used in the various network models. Besides, the depth and width of the neural network also greatly affect image classification accuracy. SENet won the ILSVRC2017 classification task championship. But CSM fuses channel features with spatial features for better representation capabilities, and CSM performs better than SENet. On the other hand, it also proves that the quality of attention model determines the final classification accuracy, which is related to the interpretability of the model. The obtained prediction density map after fusing feature self-learning idea is nearer to the ground-truth map. This phenomenon can also be observed in Fig.6, the feature self-learning loss module plays an important role in the corresponding changes of MAE accuracy.

Table 2. Results of the experiment on the ImageNet-1K dataset

Architecture	Top-1 error (%)	Top-5 error (%)
ResNet-18	29.62	10.56
ResNet-18+SE	29.43	10.24
ResNet-18+CSM	29.29	10.12
ResNet-34	26.71	8.62
ResNet-34+SE	26.16	8.37
ResNet-34+CSM	26.03	8.28
ResNet-50	24.55	7.52
ResNet-50+SE	23.21	6.74
ResNet-50+CSM	22.78	6.57
WideResNet18(widen=1.5)	26.86	8.91
WideResNet18(widen=1.5) +SE	26.23	8.51
WideResNet18(widen=1.5) +CSM	26.14	8.49

4.3. Image Classification on CIFAR-100

Based on Cifar-100, we conduct image classification experiment to verify the effectiveness of the CSM. ResNet and WideResNet are also used as baseline model. Table 3 shows the experimental results. The experimental results prove that the combination of CAM and SAM can improve classification accuracy. Besides, the depth and width of the neural network also greatly affect image classification accuracy. As we can see, our proposed method has the minimum absolute error and the minimum mean square error on CIFAR-100 dataset. Compared with other algorithm, The accuracy of the model with added attention and transfer learning strategies is greatly improved,

which reflects the best performance of the our proposed algorithm. The model is also more robust to moderately dense data sets.

Table 3. Results of the experiment on the ImageNet-1K dataset

Architecture	Accuracy (%)
ResNet-18	91.7
ResNet-18+CSM	93.1
ResNet-34	92.4
ResNet-34+CSM	93.8
ResNet-50	92.9
ResNet-50+CSM	94.3
WideResNet18(widen=1.5)	92.8
WideResNet18(widen=1.5) +CSM	94.2

As can be seen from Table **3**, the prediction density map that we got after adding feature self-learning is closer to the true value, so we can draw a conclusion that we can more accurately describe the feature through adding feature self-learning Loss reduction and corresponding changes in MAE accuracy (see Fig.7 and Fig.8), we also give the performance of net similarity.

The proposed model is evaluated with several latest models on our constructed dataset and two benchmarks, and the results are given in Tables 4 and 5. From all the tables, we notice that our method is always much better than the previous method. Table 4 gives a comparison of the ImageNet-1K, and their images are closer to the real monitoring scene than other data sets. Our proposed model has achieved considerable improvement compared to the existing technology. Table 5 indicates that proposed method obtained the best MAE and competitive MSE among the five latest methods in the CIFAR-100 dataset. As we can see, our model performs best in the most compared algorithms both in terms of MAE and MSE. The spatial complexity of the network is only related to the size of the convolution kernel, the number of channels and the depth of the network. However, it has nothing to do with the size of input data. Since the network model designed by us adopts optimization and transfer learning strategies, compared with other existing popular algorithms, the complexity of our designed model is lower.

The result of experiment is shown in Figure 7. As is shown in Figure 7, it is easily concluded that the ResNet-50 model with CSM has achieved higher accuracy. We can observe that CAM perform better than SAM. Besides, the combination of two attention modules can bring a better performance. The experiment shows that it is effective to conduct CAM and SAM at the same time. The experimental results are shown in Table 6. Based on the above analysis of the experiment, we find that it is effective to conduct CAM and SAM at the same time for improving the interpretability ability of neural networks. So, in the ablation experiment, we compare the effect of CAM and SAM in different order of use. The baseline network with CAM and SAM, baseline network with SAM and CAM are conducted respectively. Figure 7 shows the result of experiment.

Table 4. Comparison in the ImageNet-1K data set

Methods	MAE	MSE
Optimization Algorithm [44]	419	541
XAI [27]	467	498
Software-Defined [13]	377	509
Cost-Driven [8]	322	**341**
Aquila Optimizer[45]	318	439
ours	**280**	379

Table 5. Comparison in the ImageNet-1K data set

Methods	MAE	MSE
Optimization Algorithm [44]	181	277
XAI [27]	154	229
Software-Defined [13]	110	173
Cost-Driven [8]	101	152
Aquila Optimizer[45]	97	145
ours	**91**	**137**

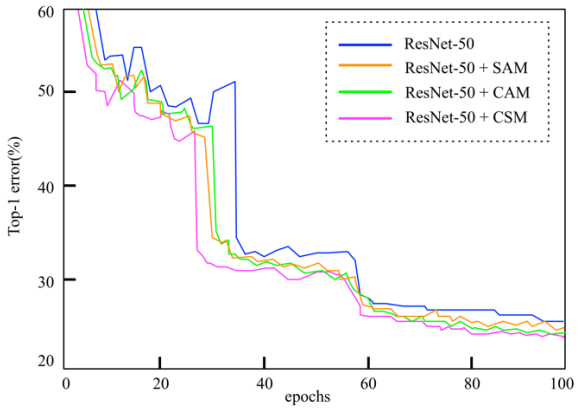


Fig. 7. Comparison of different network models

Similar to the above experiment, adding the attention module still bring improvement on image classification accuracy (as is shown in Figure 8). We can observe that the CAM-first combination method achieves better performance. The order in which models are executed can affect the accuracy, which is a supplement to the interpretability of the model.

Table 6. Comparison of different network models

Architecture	**Top-1 error (%)**
ResNet-50	24.55
ResNet-50+SAM	23.47
ResNet-50+CAM	23.21
ResNet-50+CSM	22.78

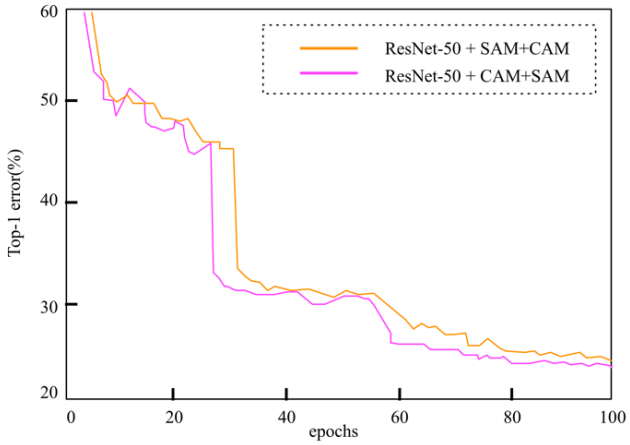


Fig. 8. Comparison of different network models

5. Conclusions

Different from the previous works on image classification, we propose proposes an explainable multi-channel convolutional neural network model consisting of channel attention module and spatial attention module. This module derives the attention map by CAM and SAM respectively. The CAM selectively enhances some feature channels and suppresses some feature channels by learning the relational mapping. The SAM aggregates features by weighting features at spatial dimension. We conducted a lot of image classification experiments for comparison on the ImageNet-1K and Cifar-100 dataset. In fact, the attention module can be well embedded in different deep neural networks and it improves the explainable deep neural network's ability of expression. Besides, the width and depth of the neural networks are also researched through self-paced learning and transferring learning scheme.

In the future research, we will pay more attention to the improvement of real-time algorithm and effectiveness of explainable deep learning, also be able to obtain more accurate data collection to develop an intelligent prediction machines for image classification based on more effective machine learning approaches.

Acknowledgment. This work is supported by the 111 Project (B12018); Innovative Research Foundation of Ship General Performance (14422102).

References

1. Diba, A., Sharma, V.: Weakly super-vised cascaded convolutional networks. In: IEEE conference on computer vision and pattern recognition (CVPR). New York, NY, 5131-5139. (2017)

2. Ren, S. H., He, K. M., Girshick, R.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Analysis And Machine Intelligence, Vol. 39, 1137–1149. (2017)
3. Ross, G., Donahue, J., Darrell, T., Malik, J.: Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: IEEE conference on computer vision and pattern recognition (CVPR), Columbus, Ohio, 580–587. (2014)
4. Chen, X., Li, M., Hsu, C. H.: DNNOff: Offloading DNN-based Intelligent IoT Applications in Mobile Edge Computing. IEEE Transactions on Industrial Informatics, 612-623. (2021)
5. Chen, X., Chen, S. H., Ma, Y., Huang, G.: An Adaptive Offloading Framework for Android Applications in Mobile Edge Computing. SCIENCE CHINA Information Sciences, Vol. 62, No. 8, 821-832. (2019)
6. Zhang, Y., Huang, G., Liu, X. Z., Zhang, W., Mei, H.: Refactoring android Java code for on-demand computation offloading. ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, 980–987. (2012)
7. Chen, X., Hu, J., Chen, Z. Y., Lin, B., Min, G. Y.: A Reinforcement Learning Empowered Feedback Control System for Industrial Internet of Things. IEEE Transactions on Industrial Informatics, 1109-1124. (2021)
8. Lin, B., Huang, Y., Zhang, J. S., Hu, J., Li, J.: Cost-Driven Offloading for DNN-based Applications over Cloud, Edge and End Devices. IEEE Transactions on Industrial Informatics, Vol. 16, No. 8, 5456-5466. (2020)
9. Lin, B., Zhu, F., Zhang, J. S., Chen, J., Mauri, J. L.: A Time-driven Data Placement Strategy for a Scientific Workflow Combining Edge Computing and Cloud Computing. IEEE Transactions on Industrial Informatics, Vol. 15, No. 7, 4254-4265. (2019)
10. Gong, C., Ding, Y., Han, B., Yang, J., Tao, D. C.: Class-Wise Denoising for Robust Learning under Label Noise. IEEE Transactions on Pattern Analysis and Machine Intelligence,12-28. (2022)
11. Gong, C., Wang, Q. Z., Liu, T. L., Hu, J., Yang, J., Tao, D. C.: Instance-Dependent Positive and Unlabeled Learning with Labeling Bias Estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence,189-199. (2021)
12. Chen, X., Zhu, F. N., Zhang, J. S., Chen, Z. Y., Min, J. Y.: Resource Allocation for Cloud-based Software Services Using Prediction-Enabled Feedback Control with Reinforcement Learning. IEEE Transactions on Cloud Computing, 111-125. (2022)
13. Huang, G., Luo, C. R., Wu, K. D., Ma, Y., Zhang, Y.: Software-Defined Infrastructure for Decentralized Data Lifecycle Governance: Principled Design and Open Challenges. IEEE International Conference on Distributed Computing Systems, 236-252. (2019)
14. Florian, S., Kalenichenko, D., Philbin, J. S.: FaceNet: A Unified Embedding for Face Recognition and Clustering. In: IEEE conference on computer vision and pattern recognition (CVPR), 815–823. (2015)
15. Leung, Y., Ji, N. N., Ma, J.: An integrated information fusion approach based on the theory of evidence and group decision-making, Information Fusion,Vol. 14, No. 4, 410-422. (2013)
16. Parkhi, O. M., Andrea, V., Andrew, Z.: Deep Face Recognition. In: British machine vision conference (BMVC), 110-119. (2015)
17. Jamie, S., Andrew, F., Mat, C., Alex, K., Blake, N.: Real-Time Human Pose Recognition in Parts from Single Depth Images. In: IEEE conference on computer vision and pattern recognition (CVPR), 1297–1304. (2011)
18. Newell, A., Yang, K. Y., Deng, J.: Stacked Hourglass Networks for Human Pose Estimation. In: Computer Vision-ECCV, 483–99. (2016)
19. Alex, K., Sutskever, I., Geoffrey, E., Hinton, J.: ImageNet Classification with Deep Convolutional Neural Networks. Communications of The ACM. Vol. 60, 84–90. (2017)
20. Cires, A., Dan, U., Juergen, S.: MultiColumn Deep Neural Networks for Image Classification. In: IEEE conference on computer vision and pattern recognition (CVPR), 3642–3649. (2012)

21. Chen, S., Gong, C., Yang, J., Niu, G., Sugiyama, M.: Learning Contrastive Embedding in Low-Dimensional Space. Annual Conference on Neural Information Processing Systems. Berlin Heidelberg New York. (2022)
22. Pentina, A., Sharmanska, V., Lampert, C. H.: Curriculum learning of multiple tasks. In Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition, 2547–2554. (2015)
23. Lin, L., Wang, K., Meng, D., Hu, J., Li, J.: Active Self-Paced Learning for Cost-Effective and Progressive Face Identification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 7-19. (2017)
24. Holzinger, A., Biemann, C., Constantinos, S. P., Douglas, B.: What do we need to build explainable ai systems for the medical domain?, arXiv:1411.1784 (2017)
25. Arrieta, A. B., Rodriguez, N. D., Ser, J. D., Bennetot, A., Herrera, F.: Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Information Fusion, vol. 58, 82-115, June (2020)
26. Xu, D., Yang, W., Xavier, A. P., Wang, X. J., Sebe, N.: Learning deep structured multi-scale features using attention-gated crfs for contour prediction. In NIPS, 3961-3970. (2017)
27. Sachan, S., Yang, J., Xu, D. L., Benavides, D., Li, Y.: An explainable AI decision-support-system to automate loan underwriting. Expert Systems with Applications, Vol. 144, 113-130. (2020)
28. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE conference on computer vision and pattern recognition (CVPR), 28-34. (2016)
29. Simonyan, K., Zisserman, A., Zhang, J. S., Hu, J., Li, J.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv preprint arXiv: 1409.1556 (2020)
30. Abhronil, S., Ye, Y., Wang, R., Liu, C., Roy, K.: Going Deeper in Spiking Neural Networks: VGG and Residual Architectures. Neuroscience. Vol. 13, 95-95. (2009)
31. Xie, S., Girshick, R., Dollar, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. arXiv preprint arXiv: 1611.05431 (2016)
32. Fei, W., Jiang, M., Qian, C., Yang, S., Li, C.: Residual Attention Network for Image Classification. In IEEE conference on computer vision and pattern recognition (CVPR), 6450–6458. (2017)
33. Yang, K. Y., Qinami, K., Li, F. F., Deng, J., Olga, J.: Towards Fairer Datasets: Filtering and Balancing teh Distribution of the People Subtree in the ImageNet Hierarchy. In: Conference on fairness, accountabiility and transparency (FAT) (2020)
34. Olga, R., Deng, J., Su, H., Krause, J., Ma, J.: ImageNet Large Scale Visual Recognition Challenge. International journal of computer vision (IJCV), 1021-1034. (2015)
35. Deng, J., Berg, A., Li, K., Hu, J., Li, F. F.: What does classifying more than 10,000 image categories tell us? In: Computer Vision-ECCV (2010)
36. Russakovsky, O., Li, F. F.: Attribute Learning in Largescale Datasets. In: Computer Vision-ECCV (2010)
37. Deng, J., Dong, W., Socher, R., Li, F. F.: ImageNet: A Large-Scale Hierarchical Image Database. In: IEEE international conference on computer vision and pattern recognition (CVPR), 1354-1361. (2009)
38. Hu, J., Shen, L., Sun, J.: Squeeze-and-excitation networks. In: IEEE conference on computer vision and pattern recognition (CVPR), 7132-7141. New York, USA. NY: IEEE (2019)
39. Zhang, Y. L., Li, K. P., Li, K., Wang, L., Zhong, B.: Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In: Computer Vision ECCV. 294–310. (2010)
40. Woo, S., Park, J., Lee, J. Y., Kweon, J.: CBAM: Convolutional Block Attention Module. In: Computer Vision-ECCV. 3-19. (2010)
41. Chen, L., Zhang, H. W., Xiao, J., Nie, L. H., Liu, W.: SCA-CNN: Spatial and Channel-Wise Attention in Convolutional Networks for Image Captioning. In: IEEE international conference on computer vision and pattern recognition (CVPR). 6298–6306. (2017)

42. Zhao, H. S., Zhang, Y., Shi, J. S., Lin, J., Jia, J.: PSANet: Point-Wise Spatial Attention Network for Scene Parsing. In: Computer Vision-ECCV. 267–283. (2018)
43. Meng, X., Huang, Y., Zhang, J. S., Hu, J., Li, J.: Feature selection and enhanced krill herd algorithm for text document clustering. Computing reviews, Vol. 60, No. 8, 318-328. (2019)
44. Abualigah, L., Diabat, A., Mirjalili, S.: The Arithmetic Optimization Algorithm. Computer Methods in Applied Mechanics and Engineering, 376-389. (2021)
45. Laith, A., Dalia, Y., Mohamed, A. E., Amir, H. J., Li, J.: Aquila Optimizer: A novel meta-heuristic optimization algorithm, Computers & Industrial Engineering, Vol. 157, 28-41. (2021)
46. Johnson, J., Alahi, A., Li, F. F.: Perceptual Losses for Real-Time Style Transfer and Super-Resolution. Springer, 120-138. (2016)

Bin Wu received the B.Sc. degree from Jiangnan University, Wuxi, China, in 1996, and the M.S. degree from Jiangnan University, Wuxi, China, in 2005. He is currently a lecturer with the School of Internet of Things Engineering, Jiangnan University. His major research interests include visual surveillance, object detection, integrated circuit design and application of embedded system.

Tao Zhang received the bachelor's degree from Henan Polytechnic University, Jiaozuo, China, in 2008, and the Ph.D degree from the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China, in 2016. He is currently an Associate Professor with the Jiangsu Provincial Engineering Laboratory for Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi, China. He has led many research projects (e.g., the National Science Foundation and the National Joint Fund), He has authored over thirty quality journal articles and conference papers. His current research interests include data mining, information systems, wireless network, artificial intelligence, IoT and security, medical data analysis, visual surveillance, scene understanding, behavior analysis, object detection, and pattern analysis.visual monitoring, scene understanding, behavior analysis, target detection and pattern analysis.

Li Mao received the B.Sc. degree from Southeast University (Nanjing, China) in 1990 and M.S. degree from Donghua University (Shanghai, China) in 2003. He is currently an associate professor at the department of computer science, Jiangnan University. His major research interests include visual surveillance, object detection, and data mining.

Received: July 14, 2022; Accepted: December 18, 2022.

Computer Science and Information Systems 20(2):765–792 https://doi.org/10.2298/CSIS220719021Z

Using Artificial Intelligence Assistant Technology to Develop Animation Games on IoT

Rong Zhang

School of Artificial Intelligence, Dongguan Polytechnic,
Dongguan, 523808, China
443798430@qq.com

Abstract. This research proposes an XNA animation game system with AI technology for action animation games in mobile devices, based on an object-oriented modular concept. The animation game function with AI technology is encapsulated into independent objects, through the combination of objects to build repetition. It adds AI technology to the finite state machine, fuzzy state machine and neural network and attempts to combine the traditional rule-base system and learning adaptation system to increase the learning ability of traditional AI roles. The main contributions are compared with traditional methods and the AI animation game system is shown to have more reusability, design flexibility and expansibility of its AI system through the object composition approach. It adds AI technology to combine the traditional rule-base system and learning adaptation system to increase the learning ability of traditional AI roles. Therefore, AI animation game producers can accelerate their processes of developing animation games and reducing costs.

Keywords: Artificial intelligence assistant, Animation games, Neural network, Object-oriented modular, Fuzzy state machine.

1. Introduction

Due to the rapid development of the global digital content industry, various countries have invested in the promotion of the digital content industry, and the animation game industry has become one of the most promising industries. The research pointed out that in terms of annual expenditure, the growth rate of television (TV) animation games is 9.6% higher than that of TV and is second only to the Internet. In terms of the average hours spent per year, TV animation games are growing and the fastest-growing media has a compound annual growth rate of 7%. Animation game producers develop games with AI technology, increasing the perception of players that they are intelligent beings, so that players can have a deeper and more interesting experience when interacting with these AI characters in animation games. From this point of view, AI animation games are not only an indispensable technology for most animation games but are also one of the sources of interactivity and fun in animation games [1, 2].

In today's digital electronic technology, the technical power of the digital animation gaming industry is growing rapidly, through the mutual promotion of software and hardware. The speed of the improvement of animation game graphics is obvious in all

game fields. In the early 2D surface to the mid-term 3D stereo, and to the next generation of high-resolution images, animation games are constantly progressing in terms of screen effects that continue to shock players. AI research experts claim that when animation game screens improve, animation games are not promoted in the same way [3, 4]. However, due to the advancement of graphics technology, animation game characters have smoother movements, more realistic textures and more complex expressions. If they do not have an effective AI animation game to control their behaviour, the animation game characters will be deemed as not having a real appearance, and instead will be characterized as having asymmetric and strange behaviour [3, 4]. Therefore, as regards superior animation games, AI is a very important technology for today's animation games. In the past, animation game development used DirectX for programming, however, DirectX did not provide much of a design function to animation games, therefore, the designer was responsible for a large number of implemented parts. The XNA animation game is an integrated cross-platform development environment for PC/XBOX mobile animation games and includes an animation game development library, packaged software tools and various other development tools, so as to provide the most convenient and efficient development environment for AI animation game developers. It brings together 3D drawings, physics engines and animation game production process management, as well as resource management and other applications to help AI animation game producers accelerate the development of animation games and reduce costs. Fig. 1 is a diagram showing the relationship between XNA and animation games. In Fig. 1, XNA is an animation game function library, which provides the basic functions of animation games and an empty animation game architecture. The designer only needs to expand the functions according to its architecture [5, 6]. The second layer is the animation game engine, designed by the designer to expand and establish the basic category library and functions, according to the functions library required by the animation game. The third layer is the animation game code, using the first and second layer category libraries, and the function is used to write the code for the actual operation of the animation game. In the case of a small animation game, the designer can even skip the second layer, as long as the function provided by XNA is used directly to write the animation game program in an empty animation game; this can make the animation game work. Based on the aforementioned advantages, XNA provides a very good development environment for amateur animation game designers and students [7, 8].

The related AI animation game technology research can be roughly divided into: chess AI animation game implementation, AI animation game implementation, based on the open architecture of existing animation games and self-made AI animation game system implementation [9, 10]. Most AI animation game technology researchers will choose animation games with simpler rules to realize the idea of AI technology. Nowadays, many commercial animation games openly authorize players to modify part of the animation game program. Players can implement their own ideas in the animation game, according to the open modification framework of the game. This has also led to research such as the AI implementation of animation games, based on the open architecture of existing animation games [11.12]. This research uses open-licensed commercial animation games to modify programs, so as to realize researchers' AI technical ideas, although researchers can quickly use such animation games to implement AI animation games. However, these will also be subject to the original

architecture of animation games, therefore, the AI technology that can be implemented will be limited. Although the AI system implementation of self-made animation games needs to build an animation game system independently, it gives researchers considerable freedom and flexibility. The first objective of this research is twofold. Firstly, the aim is to implement action animation games with XNA in the AI system. Through the concept of object-oriented modular objects, an AI system is built with reusability, design flexibility and expandability. In addition, finite state machines, fuzzy state machines and neural network AI technology applications are added, combined with traditional rule-based systems and learning adaptation systems, so that traditional AI roles can increase their learning capabilities [13, 14]. The second purpose is to show that XNA can simplify the characteristics of animation game design through the process of implementing the AI system; it can explain and record the implementation process to provide a reference for related researchers and producers. This research sets the implementation goal as the AI system implementation of action animation games. By focusing on the AI system implementation of a single animation game type, the results of the research purpose can be achieved within the expected timeframe [15, 16].

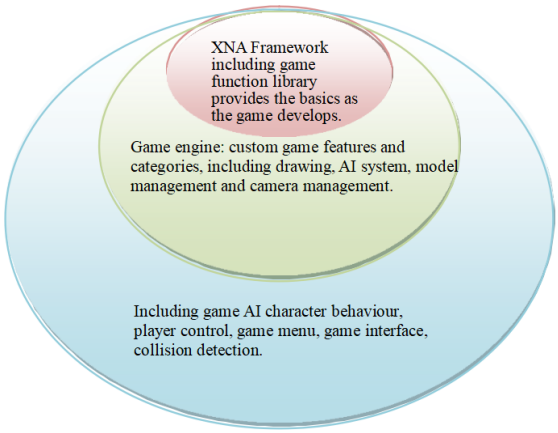


Fig. 1 The relationship between XNA and animation games

Although the AI animation game is implemented using an academic concept, issues, such as animation game execution efficiency must also be considered. Therefore, an AI animation game usually simplifies the complex algorithmic process or only uses its concept to implement an AI system, as its main purpose is to perform the desired AI effect under the allowable performance consumption. This research will focus on the AI system used in animation games, with the aim of demonstrating that the research results can be more in line with the current animation game production flow process, so as to improve the performance value of the research results. Animation game production can be roughly divided into three areas: planning, art and programming. Each area constitutes a different professional field, however, this research will only discuss the programming element of AI animation game production [17, 18].

The first section discusses the research background and motivation, as well as the research purpose and scope of the research. Section two discusses related background

knowledge and the AI performance that is emphasized in the basic types of animation games. Section three focuses on the system planning and analysis of AI animation games and explains the system architecture. Section four considers the implementation of AI animation game systems and explains in detail the function and structure of each object. Section five discusses the implementation, which through combination and connection becomes a working example implementation; this section also highlights the reusability, design flexibility and scalability of the AI system. Section six comprises the conclusion and explains the research results and the future work.

2. Related Work

2.1. Animation Game Types and AI Performance

The type of animation game greatly affects the function and direction planning of the AI system. Different types of animation games focus on different AI performances. Therefore, AI systems are rarely universal and are usually built for specific animation games. There are many types of animation games today. Table 1 lists the common basic animation game types. In addition to the basic types in the table, there are also many mixed types, such as the Action Role Playing Game, ARPG) and the Action Adventure Game (AAVG), [19, 20].

The basic animation game types listed are based on these animation game types. The following key AI performances are summarized [21-25]:

1. Fixed behaviour: the behaviour of a fixed mode does not change according to the time of the animation game. The designer defines a number of behaviour patterns and conditions that trigger behaviours and the AI animation game characters make corresponding behaviour patterns, based on the defined conditions. They do not change in line with the time of the animation game or any other factors.

2. Variable behaviour: the behaviour of the non-fixed mode will change in accordance with the time of the animation game or other factors. The designer also defines several behaviour patterns, but the conditions that trigger the behaviour will change, according to the time of the animation game and other factors.

3. Simulation behaviour: AI characters record and simulate the player's good performance in animation games.

4. Group action: this is the behaviour pattern of AI characters in group actions of multiple units. For example, in a real-time, strategy animation game, when a large number of AI animation game units are moving, the group action will form a melee unit in front and a long-distance unit behind or teamwork behaviour in basketball or football animation games.

5. Strategic thinking: this constitutes AI management performance that belongs to high-level decision makers and supervises and collects various information in animation games at a higher position. It integrates the information to make decisions and informs single or multiple AI animation game units of the decision.Nowadays, the number of animation games is quite considerable and each animation game may contain multiple

AI performances. Therefore, Table 2 mainly focuses on the most common AI performances of this type. It shows that the AI performance of action animation games has only one fixed behaviour, although role-playing animation games and adventure animation games are also in the same situation. The difference is that the objectives of the other three types of animation games are not only focused on defeating enemies. For example, role-playing animation games also focus on character training and adventure animation games focus on solving puzzles, dodge mechanisms and the process of jumping on to a platform [26-30]. The goal of the action animation game is to defeat the enemy. Therefore, this research aims to enhance the player experience of action animation games, by enriching the reactions and actions of AI characters. In order to achieve this goal, the research will focus on AI characters to add changing behaviour. The AI animation game is the broadest definition, including everything from simple chasing and dodging movement modes to neural networks and genetic algorithms. Using AI to allow non-player characters to show different personality traits or to present human-specific emotions and tempers, is one of the options that can be considered in the design of animation games [31-35]. No matter what it entails, as long as it can give people a certain level of wisdom, make animation games more addictive, more challenging and importantly fun, it can be regarded as an AI animation game.

Table 1. Common types of animation games available currently

Type of animation game	Feature description	Famous animation
Role Playing Game	Focuses on plot description and character development. Usually has a broad, complete world view and gives attention to the player's sense of investment in the world of animation and game.	Final Fantasy[12]
Adventure Game	Comprises a puzzle element and players must solve through the puzzle to pass through the levels.	Tomb Raider [6]
Real-Time Strategy	Players need to control a large number of animation and game units at the same time to fight, and the animation and game unit action is immediate.	Age of Empires[11]
First-Person Shooter	A first-person perspective of the animation game; the players shoot as the main means of attack.	Call of Duty
Platform Game	There are a lot of units and organs on the map. Players must jump to avoid the organs, so as to play the animation game.	Super Mario [15]
Sports Game	Simulation of various sports, such as ball games, swimming, skiing, etc.	NBA Live
Racing Game	This game has a sense of speed, such as racing cars or racing boats.	Need for Speed
Action Game	A single character completes a level by knocking down the enemy. The action is instantaneous.	Ninja Gaiden
Fighting Game	Fights a single enemy at a time, using a variety of knockouts.	Street Fighter
Table Game, TAB	Realistic desktop animation game simulation, such as chess, monopoly, Mahjong, etc.	Monopoly

2.2. AI Animation Game

AI animation games are divided into three categories, namely rule-based systems, goal-based systems and learning and adaptation systems. The following relates to the rule-based system and the learning and adaptation system, which are more commonly used in the development of animation games.

1. Rule-based system: this type of system has a long history in animation games and also uses the most common AI technology in animation game development. Its operating principle involves the designer defining the conditional rules corresponding to the behaviour of AI characters in advance, then using the programming language The conditional judgement (if, then, else, switch, case) is used to control the behaviour of AI characters. The advantage of this type of system is that it is simple and easy to understand as well as to debug and implement, however, its disadvantage is that it lacks learning ability and is easily predicted by players. There are three common architectures for such systems, namely the finite state machine (FSM), fuzzy state machine (FuSM) and production systems.

2. Learning and adaptation system: as the name suggests, this type of system has the ability to learn and adapt. It can make AI characters more changeable and challenging in animation games since players cannot easily predict the behaviour of AI characters. Due to the continuous improvement of AI characters, the entertainment life of animation games has also been extended. According to the AI learning technology used, this can roughly be divided into two categories, namely indirect learning and direct learning.

1. Indirect learning: indirect learning is a mathematical statistical method combined with a rule-based system. It counts the player's control patterns and habits, and analyses them, then uses the analysed data to modify the AI character's behavioural rules, so that the AI character has the ability to learn and adapt to the player [36, 40].

2. Direct learning: direct learning is based on AI technology with learning concept algorithms as the system architecture, allowing AI characters to have autonomous learning capabilities through learning algorithms. In this type of method, the designer usually establishes a learning system first, then creates a system with learning capabilities through different training methods. AI characters in animation games use the trained system to conduct behavioural, decision-making analysis. An AI character, built entirely using this type of system has strong learning and adaptability capabilities, but this will also cause problems for animation game developers, such as difficulty in debugging or the AI may deviate from the expected effect. Moreover, such systems, which usually contain complex algorithms are difficult to understand and implement. There are two common applications of learning and adaptation systems in animation games, namely the artificial neural network and the genetic algorithm [41-45].

Table 2 below compares the advantages and disadvantages of the two AI systems and shows that the advantages and disadvantages of the two AI systems have complementary effects. If only one single type of AI system is used in an animation game, when the animation game is scaled, the larger the size, the greater the number of shortcomings. Therefore, in current animation game development, different functions are usually established for different needs. Either a subsystem of power is established or the AI work is divided into several small systems, then multiple subsystems are combined to form a complete AI animation game system, so as to maximize the advantages and minimize the shortcomings. Consequently, the AI system modularization of the key

technology of the AI technology is very important. The modularized AI technology will be reusable in the system and can effectively combine the subsystems. The meaning of modularization is reusability, which can improve flexibility [45].

Table 2. Comparison of the advantages and disadvantages of the rule-based system and the learning and adapting system

	Advantage	Disadvantage
Rule-based system	• Easy to understand. • Easy to implement. • Debugging is easy.	• The larger the animation game scale, the more the program volume will lead to confusion. • Predictable.
Learning and adaptation system	• It is learned. • It is hard to predict. • It simplifies complex programming.	• Implementation is not easy. • Debugging is difficulty.

2.3. FSM

FSM: also known as finite state automata or state machine for short, this is a mathematical model that represents a finite number of states and behaviours, such as transitions and actions between these states. A state machine usually contains three elements: the transition functions between all states, inputs and connection states in the state machine [21-25]. The FSM is used in the AI program of animation games. The AI chasing the player's enemy in the animation game is the application of the FSM and this is used in the AI of animation games. The reason for this is that it easy to understand, implement and debug characteristics, and its characteristics also make finite states. In the process of developing animation games, the machine often has outstanding performance. In the following section, the enemy AI in the classic animation game, Pac Man, is used as an example to explain the use of FSMs in animation games [46-50].

The FuSM is actually an extended variant of the FSM. These statements are closer to everyday problems and semantic statements, and examples are used to illustrate the concept of fuzzy logic. In an animation game, there is a castle and according to the concept of classic logic, when the character enters the castle, it is 1 and not in the castle. The inner time is 0, which makes the character's behaviour in the animation game more flexible and realistic. The difference between the FuSM and the basic state machine is that the FuSM allows multiple states to run at the same time, while the basic state machine can only run one state at a time (Kwon and Shin, 2005). The parameter can be used as the stimulus value of the FuSM state. Firstly, the stimulus value standard for triggering each state, when the enemy is close to the point, may result in the FuSM operating in the three states of moving, shooting and dodging at the same time.

The production system can also be called an expert system and combines the functions of a knowledge database, rule management and decision-making. It is usually used to solve expert problems in a specific field. In the production system used by AI characters in animation games, the users of the system are the AI characters and the process of operating the system used by the AI characters is automatically run by the

computer, so there is no need to provide a user interface to the AI characters. Animation game programmers usually write scripting languages corresponding to the AI system and provide these to animation game planners to design AI animation games [26-29]. The production system is famously used in animation games and the main AI system is established by the production system. The computer AI in animation games can learn the building order and the resource requirements of the building through the knowledge base, then use the reasoning mechanism to determine the search for resources and collection acts [51-54].

3. Research Design

3.1. AI System Construction Process

The scope of the AI animation game is quite large; complex human thinking simulation to path search are all within the scope of the AI animation game. Therefore, before building an AI system, one must first determine the main use of the AI system, such as purpose, expected effects or information relating to animation games, then the function and architecture of the AI system can be planned, based on the established information content. Finally, the system can be implemented based on the planned content. The AI system construction process can be divided into three periods, the system direction and basic content setting period, the AI system planning period and the AI system implementation period:

1. System direction and basic content setting period: the key function of the AI system builders is to confirm the animation game type, animation game content and the expected performance goals of the plan, as well as to carry out the preliminary planning of the system, based on the information obtained and to analyse and select the required software.

2. AI system planning period: the main work content is to carry out detailed planning of the AI system, including the AI system architecture, the application scope of AI technology, the information required by the AI system, the operation method of the AI system and the animation games to connect to the main structure.

3. AI system implementation period: this relates to the AI system construction work, according to the content planned in the AI system planning period, and the results are realized by program software. In addition to the implementation of the system, this issue also includes system testing and problem modification.

3.2. Software and Hardware Requirements

(1). Software: Visual C# was developed to realize the most efficient programming language in the .NET Framework. Visual C#, like Java, is an intermediate code language. The base language used by XNA is C#, so the main language used in animation game programming is Visual C#.

(2). Hardware: the software used in this study is XNA Game Studio. Therefore, based on the hardware requirements of XNA, it is recommended to have at least 4GB of physical memory and a CPU clock of 2.0GHz or more.

3.3. AI Animation Game System Design

The AI animation game system is a subsystem built on animation game architecture. The AI system cannot operate independently because the AI system itself does not include graphics, model management, cameras and other animation game functions. It must be combined with other units of the animation game architecture. Therefore, for the AI system to function normally, it must have a foundation architecture in the main body of the animation game. It uses XNA as the animation game development platform, therefore, XNA will be used as the bottom layer of the animation game's main structure for planning purposes. When creating XNA's new animation game project, XNA will provide designers with an empty animation game architecture and designers must follow this architecture to expand and extend animation games. Fig. 2 shows that this architecture contains five main functions, which are Initialize, LoadContent, Update, Draw, UnloadContent, when the animation game starts; Initialize and LoadContent will be executed in sequence. After running Update and Draw, these two functions will be executed repeatedly during the animation game process until the animation game ends and UnloadContent will only be executed before the end of the animation game. The five functions will be introduced as follows:

1. Initialize: The first part executed when the animation game starts, the function is to initialize various animation game data.
2. LoadContent: this function is used to load the external data required by various animation games, such as animation game models, text files, pictures, videos, etc.
3. Update: the main loop of the animation game will be continuously executed during the animation game process to update the content and information of various animation games.
4. Draw: similar to Update, Draw will be executed continuously during the animation game process and its function is to update the animation game screen.
5. UnloadContent: this is used to release the memory space used before the animation game ends.

XNA provides a sample program function, which allows designers to quickly create an object that includes the infrastructure; the operation process of this object is also the same as the main structure of the animation game. Fig. 3 is an animation game architecture diagram, established in accordance with the animation game functions required by the AI animation game system. The function descriptions of each object form part of the animation game architecture.

3.4. The Concept Architecture of the AI Animation Game System

AI is a computer technology that simulates human thinking reactions, judgement logic, and learning ability, since the purpose of AI animation games is to provide animation

game characters with a vivid performance, allowing players to believe that the characters are alive, enhancing the animation game play and challenge of animation games and making it easier for players to become absorbed in the animation game content. At the same time, animation game development must also consider its performance. A poor performance of animation games will affect players' acceptance of animation games. The AI animation game system architecture divides the characteristics of AI into the following four key points, so as to simulate human thinking behaviours: sensation, memory, thinking and judgement, action. These four key points will be explained as follows:

1. Sensation: this is referred to as information reception and represents the feelings of animation game characters in relation to the animation game world, for example, sight or hearing.

2. Memory: this is known as information access and represents the information that an animation game character must record in the animation game world, such as player information, including player level and equipment or the attack habits of the player.

3. Thinking and judgement: this is central to the AI animation game system. It represents all the action benchmarks of the animation game characters and guides the animation game characters in terms of the judgements they should make when they encounter a certain situation. Strong AI animation games can become more humane and unpredictable, based on the information obtained.

4. Action: after passing the feeling and thinking judgement, the system will analyse a relative response to the current situation and instruct the animation game character to take action. Usually in animation games, one or more actions are set for animation game characters in advance, such as attack, escape, rest, patrol, gathering, etc., and animation game characters are analysed and judged by the AI animation game system, according to the situation.

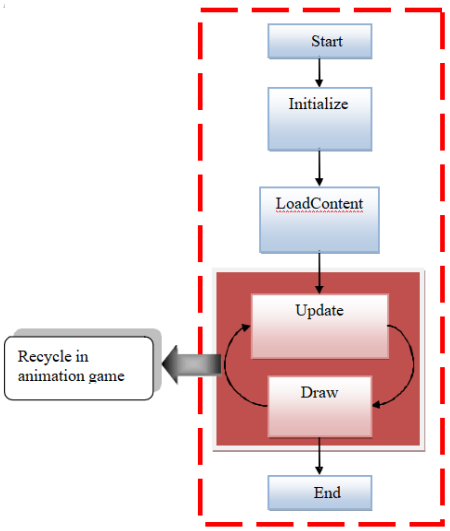


Fig. 2 Diagram of XNA animation game process architecture

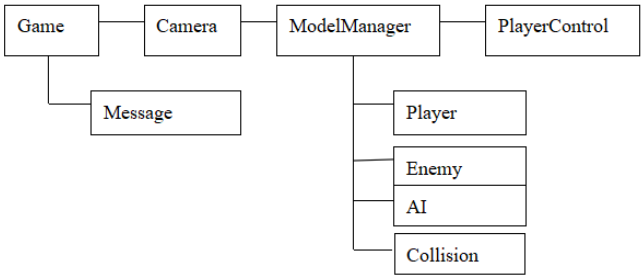


Fig. 3 Diagram of animation game architecture

In order to respond to various needs in animation games, there may be more than one AI system at the same time. These systems can operate separately or independently, or they can be related to one other.

3.5. AI Animation Game Information

The AI animation game system collects the information that is available or required in the animation game. Collecting information forms the basis and the start of the AI system. Without sufficient information, the AI system cannot create more effective and changeable thinking and judgements. Too much junk information will increase the calculation of the AI system and affect the performance of animation games, therefore, the setting and control of animation game information is very important. This information should be provided to the AI system and the type of information must be planned when the system is constructed. The animation game information used in the AI animation game system is divided into four categories.

1. World information: this type of information represents the character's information in the animation game world, such as the character's position, direction and height in the animation game world.

2. Character information: this type of information represents basic information relating to the character, such as strength and blood volume. The basic information of the character is used as a calculation basis for other information. For example, power will affect the attack damage value of the character; the higher the character, the higher the damage value it can cause.

3. Action information: this type of information not only refers to the attacking moves of the character, but also refers to the various actions of the character in the animation game, such as dodging, moving and attacking. This type of information transfers the player's current action information to the AI character, so that the AI character can determine the corresponding action. For example, when the AI character learns that a player is making an attack, the AI character may use a dodge action, when the player is on the move or on standby.

4. Situation information: this kind of information is usually used to indicate a certain situation relating to the AI character or to the player. When a player is attacked, there will be a period of time when the player cannot be attacked again, therefore, this

function must provide the collision detection system with information as to whether a character has collided with something.

4. AI System Implementation

4.1. 4.1 Main Program of Animation Game

The AI system is the main source, therefore, the main program of the animation game is built to provide the AI system with operation and testing functions. As shown in Fig. 4, the main program of animation game is divided into five parts according to its functions, and each part contains one or more categories.

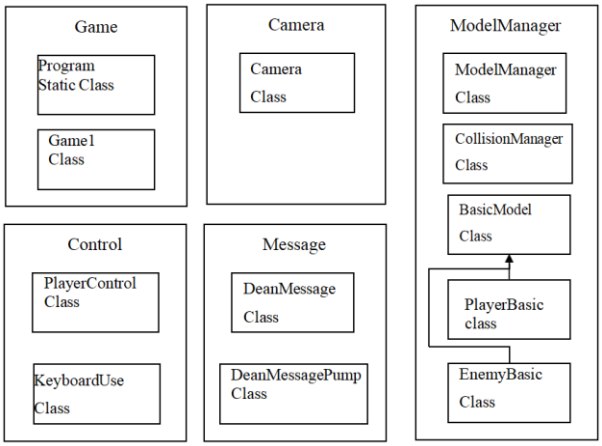


Fig. 4 Five major parts of the main program of the animation game

Object-oriented Programming (OOP) is an important concept of modular programming. An object is a combination of data and functions and the advantage of the modular object is that it can be repeated. OOP comprises three concepts, encapsulation, inheritance and polymorphism, which are explained as follows:

1. Encapsulation: to combine data, data processing procedures and functions into objects. The definition of an object in C# language is a class, which belongs to an abstract data type. To define an object is to define a new data type for the programming language.

2. Inheritance: this is the reuse of objects. When a new category is defined, it can inherit the data and methods of other categories; the new category can also add or replace the data and methods of inherited objects.

3. Polymorphic: if a category needs to process a variety of different data types, there is no need to create exclusive categories for different data types. This can directly inherit

the basic category to create a method of the same name, so as to process different data types.

(1) **Game**: this is the skeleton part of the animation game; its main function is to control the animation game process and set the animation game environment. Game contains two categories, Program and Game1, which are automatically generated by the system when a new XNA animation game project is created. The Program and Game1 categories will be described separately below:

1. Program: the function of the Program is quite simple, as it only contains a main function. This function is the first to be executed when the animation game starts to run. The main function declares an object of the Game1 category and executes the Run function of this object.

2. Game1: Game1 was originally the main loop function of the animation game; the main function is the initial setting and process control of the animation game. However, since the function of the template program has been used to allow other animation game components to operate independently, only the initial animation is left in Game1. The game environment has the function of building other animation game elements.

(2) **Camera**: the main function of the camera is to set and manage animation game cameras. It only contains a camera category, which is an animation game component, so it will update its status with the animation game time. Drawing in a 3D animation game world is like using a camera to record a video. The drawing of objects on the screen needs to be imaged through the parameters provided by the camera. The key camera parameters that affect imaging are the two matrix parameters, namely the view matrix and the projection matrix. The view matrix determines the position, facing direction and orientation of the camera in the animation game world. The projection matrix is used to determine the angle of view and the distance of the camera's visibility. Generally, the projection matrix does not need to be changed to its default value, unless the purpose is to change the visibility of the camera. The view matrix that needs to be constantly changed in animation games is determined by the three parameters of the camera position, the camera's gaze target point and the camera's upward vector. By substituting the three parameters into XNA, the CreateLook method in the function library can obtain the view matrix of the camera. If the direction of the camera is unknown in relation to the target, the player should add the camera position and the camera's facing direction to ensure that the camera is facing the target point.

(3) **ModelManager-animation game model manager**: 3D model objects are the foundation of 3D animation games. For example, animation game scenes and animation game characters, etc. are all created by 3D model objects. The function of the animation game ModelManager is to centrally manage all animation game model objects. This contains three main categories of objects, BasicModel (PlayerBasic and EnemyBasic are inherited from BasicModel), ModelManager and CollisionManager. The three categories are introduced below. BasicModel represents the basic model object category in animation games and other model objects will inherit from it. The purpose of establishing BasicModel is to unify the parts shared by all model objects in the animation games. This approach can avoid the issue of redefining the model objects each time, as shown in Fig. 5. When creating a new model object, allow it to inherit the BasicModel, then, the new model object can have the basic data and functions of the BasicModel and expand them as required. For example, PlayerBasic is the extension of

the model object representing the player and EnemyBasic is the extension of the model object representing the enemy.

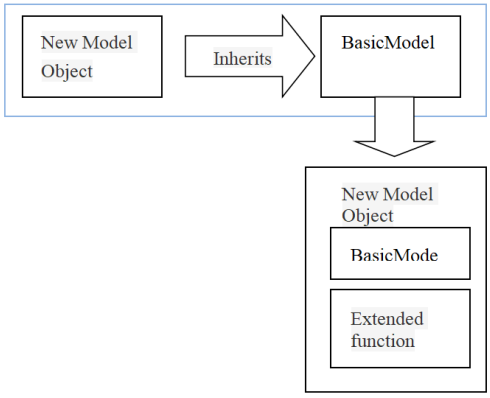


Fig. 5 A new model object inherited from BasicModel

4.2. AI system Program Description

The AI system of this study uses three AI technologies: FSM, FuSM and neural network-like technology. Since the FSM and the FuSM are quite similar in terms of basic architecture, these are two technologies which are built in the same module at the time.

(1) State machine module: this kind of FSM has two advantages. The first is high scalability. Nowadays, the scale of animation games is getting larger and the subsystems included in animation games are increasing. The main body of the AI systems of the animation game may be developed by two groups of people at the same time. In this case, the use of information becomes the communication method between the subsystems, making the subsystems independent, therefore, there is no need to worry about the information connection between the subsystems. The second point is to improve the efficiency of the system, as most AI characters are reactive. Usually, there is a change in the animation game that affects the behaviour of the AI character. Even if the AI character's behaviour does not change, the program will continue to run all the judgement programs, which will reduce efficiency. If the message appears, it is necessary to send the relevant message to the AI system when the animation game changes. The message-driven FSM uses messages as the driving force for transitions between states. In the AI system architecture, the AI manager is responsible for the decision-making element, while the state machine manager actually controls the state transition actions and state operations.

(2) Neural network module: this type of neural network can be divided into the feedforward network and the recurrent network. The characteristic of this type of network is that each layer can only pass forward and the difference between the recursive network and the feedforward network is that the recursive network allows the

network layer to pass back. In the AI design of animation games, most people choose to use the feedforward network. The reason is that the feedforward network is easier to understand and test than the recursive network, and in terms of system performance, due to the feedforward network, the path does not resemble a recursive network that requires multiple circumventions in the network, therefore, the system performance of the feedforward network is also improved. Based on these reasons, this study chose to use a three-layer feedforward network, which is divided into three network layers as the name suggests, namely the input layer, the hidden layer and the output layer.

After choosing the type of neural network, the input data, activation function, output data and training method must be planned:

1. Input data: for the neural network of the AI animation game system, the amount of input data can be controlled to a minimum, as the relatively small volume of input data also means that the neural network has a higher execution rate; too much data are surplus to requirements. The input data may cause a gap between the output of the neural network and the prediction.

2. Activation function: the activation function converts the total input value of the neuron into the output value of the neuron. The activation function used in this study is the most commonly used logistic function.

3. Output data: the output value of the similar neural network is usually the decision value that represents the expected network prediction. Since the activation function used is a logistic function, the output value will be between 0 and 1, but the actual usable range is between 0.1 and 0.9, so a value around 0.1 can be regarded as inactive and a value around 0.9 is active. In this way, provided that the output value is judged as active, it is possible to determine the decision of the neural network.

4. Training method: the purpose of training the network is to identify the weights of the interconnections between neurons, so that the input can achieve the expected output. There are two types of training methods commonly used in animation games, which are supervised learning and unsupervised learning. Unsupervised learning is independent training and does not require design. The author manually edits the training set, but the required technology is more sophisticated and more extensive, considering the difficulty and time factors, so this research chooses the easier-to-implement back-propagation method in supervised learning. Training is an optimization process, and the optimization method used by the inverse transfer method is to use the error method to minimize the error. This process can be divided into the following five steps.

Step 1: establish a training set containing input data and expected output values.

Step 2: set the initial value of the weight of the neural network to a random small value.

Step 3: pass the input data of the training set into the network and allow the network to calculate the output value.

Step 4: compare the output value calculated by the network with the expected output value and calculate the error between the two.

Step 5: adjust the weight value to reduce the error and repeat the process until the error is within an acceptable range.

In the optimization process, after calculating the cumulative error of all the input data in the training set and the expected output value, a judgement is made as to whether this has reached the acceptable range, and if not, the process is repeated. The formula for calculating the error is represented by formula 2 and 3, which are used by the output

neuron and the hidden layer neuron respectively; the input layer has no so-called error because the value of the neuron is given by:

$$\delta_i^{\ o} = (n_{di}^{\ o} - n_{ci}^{\ o})\, n_{ci}^{\ o}(1 - n_{ci}^{\ o}) \quad (2)$$

$$\delta_i^{\ h} = (\Sigma_{\delta j}^{\ o}\, w_{ij})\, n^{h}(1 - n^{h}) \quad (3)$$

where δ^o is the error of i-th output neuron in the output layer. $n_{di}^{\ o}$ is the expected output value of the i-th neuron in the output layer. $n_{ci}^{\ o}$ is the calculated output value of the i-th neuron in the output layer. δ h is the error of the i-th output neuron in the hidden layer. δ o is the error of the j-th output layer neuron, connected by the i-th hidden layer neuron. w_{ij} is the connection weight between the i-th hidden layer neuron and the j-th output layer neuron. n^{h} is the calculated output value of the i-th neuron in the hidden layer.

5. Implementation

The examples of AI system implementation are used to demonstrate the effectiveness of modular AI systems.

5.1. Example 1

Example 1 will use modular animation game objects to build a working AI character example. The following implementation process from scratch is explained in a series of steps and the implementation process is divided into three parts: pre-work, setting the main body of the animation game and establishing the AI system. The pre-work involves the process of creating a new animation game project and placing modular objects. The main body of the animation game is established to connect various functional objects of the main body of the animation game and to establish the animation game flow. Finally, the establishment of the AI system builds the corresponding AI function objects according to the set AI character reactions and actions, and connects them with the main body of the animation game.

(1). Pre-work: i) the first step is to create a new XNA animation game project. The newly created animation game project only contains Game1 and Program categories, and currently, the Game1 category only has a basic animation game empty structure without any content.

ii) the second step is to place the module object: after creating the animation game project, the main body of the animation game and the AI system module objects that have been made previously, are placed into the project.

(2). Setting the main body of the animation game: at present, the objects are still in separate conditions, therefore, it is necessary to start writing programs to connect the main parts of the animation game and to create players and AI characters. At this time, the AI characters are not connected to the AI system. The following describes the key programs in each object.

i) Game1: the most important function of Game1 in an animation game relates to the initial components. The animation game components that need to be created are Camera, ModelManager and PlayerControl. Therefore, these three objects should be created at

the initialize function stage and added to components, so as to connect them. The main loop of the animation game begins to operate. After the animation game components are established, the associations between the components must be connected. In the SceneInitialize function, the player character should be the target of the Camera, then the player character and the AI character in the ModelManager should be linked to the PlayerControl.

ii) ModelManager: creates model objects of players and AI characters in ModelManager, even though there is only one model object for both the player and the AI character. The module objects of the player and the AI character are created, then the state and drawing are updated with the ModelManager.

iii) PlayerControl: there are three kinds of player behaviours set in animation games, moving, attacking and dodging. The update function of PlayerControl will respond to the player's input every time it is updated. If this is true, the action will be executed; actions are restricted in order.

(3). Establishment of AI system: before establishing an AI system, the kind of AI roles to be established must be determined. The example below shows the creation of a close-range attacking AI character that includes fixed behaviours and variable behaviours. The AI roles linked to these behaviours are shown in Table 4.

Table 4. Example 1 Action behaviour of AI characters

AI character	Behaviour description
Patrol	Without encountering a player, the AI character performs a movement behaviour centred on the spawn point.
Chasing the player	When the player is within visual range, the AI character pursues the action of approaching the player.
Combat Movement	An AI character moves around a player-centred point while within combat distance.
Attack	When within combat distance with the player, the AI character carries on the attack action to the player.
Dodge	A fast lateral move performed by an AI character when he or she is under attack or needs to move quickly.

AI characters have five behaviour modes, which can be roughly divided into two parts: combat and non-combat. The combat element includes combat movement, attacking and dodging; the non-combat element patrols and chases the player. According to the aforementioned classification, the battle element is planned as a variable behaviour and the non-combat element as a fixed behaviour; the fixed behaviour uses an FSM and the variable behaviour uses a neural network. The plan is shown in Fig. 6.

EnemyAIManager: the AI manager connects the AI role and the AI system. Firstly, a category EnemyOneAIManager is created that inherits the EnemyAIClass, connects the EnemyOneAIManager in the EnemyOne category and updates the AIManager in EnemyOne's update function. As a result, EnemyOneAIManager is connected to the FSM. EnemyOneAIManager wishes to manage the condition information required by the state transition of the FSM and the input information required by the neural network. FSM: the connection between the AIMachine of the FSM and each state has been completed in EnemyOneAIManager. The following describes the two states of

EnemyChaseState and EnemyPatrolState; EnemyFightState uses neural networks internally.

EnemyPatrolState: this state controls the AI character's patrol action centred on the spawn point. After entering this state, the character will first calculate the distance between the current position and the centre of the circle, and will store it in the length parameter.

EnemyChaseState: this state controls the behaviour of the AI character chasing the player. The operation of this state is quite simple. Provided that the AI character is in this state, it will continue to move with the player as the target, until it leaves the state.

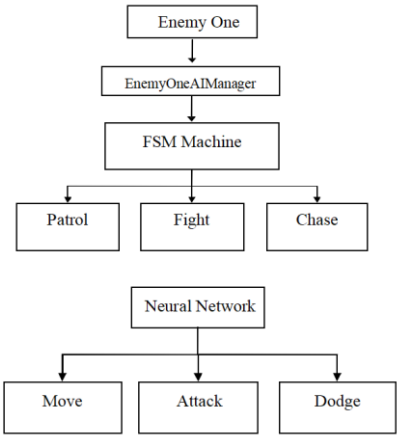


Fig. 6. Example 1 AI system planning

(4) The establishment of the AI system-neural network: the neural network in Example 1 is used to decide the combat behaviour of AI characters, so the neural network is placed in the EnemyFightState state of the AI character for use. There are three combat behaviours in the EnemyFightState state: move, dodge and attack. These three behaviours represent the output of a neural network, using input AI character emotions, current player actions and AI characters relative to the player. The three pieces of information relating to location use neural networks to determine the combat behaviour that should be executed, as shown in Fig. 7.

Before using the neural network, it is necessary to train in the neural network. The training method used here provides a supervised learning of the training set. The method needs to create a training set containing input and expected output data, as shown in Table 5.

The training process involves passing the input of the training set to the neural network to run, then comparing the obtained output with the expected output value in the training set to calculate the error. Then the inverse transfer method is used to update the weight of the input and the hidden layer; this process is repeated until all the data in the training set are calculated once, then, a judgement is made as to whether the accumulated error meets the expected value. If this is not the case, the training process is repeated until the error reaches the expected target. It takes time to train a neural network. If a large number of such training processes are repeated in an animation game,

this may cause delays in the animation game. Therefore, the neural network is built in the EnemyOne category to make the animation game in the first place. When it starts to read AI characters, the neural network must be trained first and the data stored externally, then the trained neural network data are read from the outside when the animation game is executed.

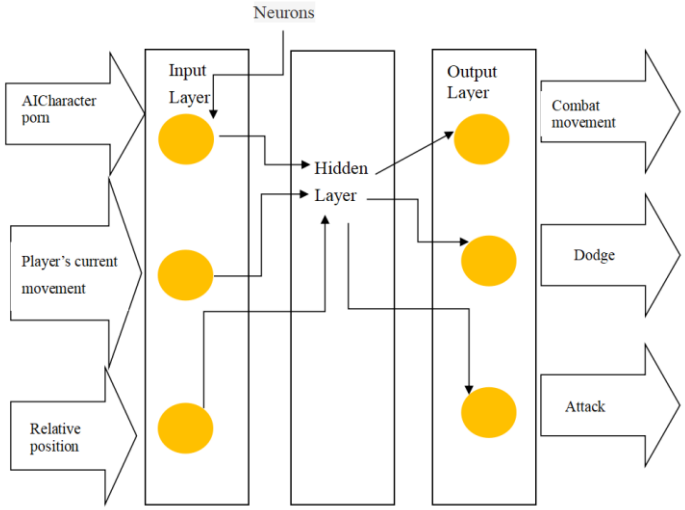


Fig. 7. The neural network in the implementation example 1

Table 5. Examples of training sets of neural networks

Input			Output		
AI character emotion	Player's current movement	Relative position	Combat movement	Dodge	Attack
0.0	0.1	0.1	0.1	0.9	0.1
0.1	0.2	0.1	0.9	0.1	0.1
0.3	0.1	0.4	0.1	0.9	0.1
0.5	0.1	0.2	0.1	0.1	0.9

Once the neural network has been built, the three combat behaviour functions, namely, move, attack and dodge are explained below.

1. Combat movement: the behaviour of move control is to allow the AI character to circle around the player. When the player approaches, the AI character moves back to keep a distance from the player.

2. Attack: the behaviour controlled by attack is an attack action. When the distance between the AI character and the player is greater than the attackable distance of the AI character, the AI character will move closer to the player and the attack action program will be executed when the distance is less than that.

3. Dodge: the behaviour controlled by dodge is the act of dodging. When the dodgeAnim parameter is false, the current position of the AI character should first be

saved to the prePosition parameter and dodgeAnim set to true, to avoid changing to prePosition. When the canDodge parameter of the AI character is true, this means that the AI character is currently able to perform a dodge action. The distance between the AI character's current position and the location to which it intends to move is judged, then the dodge action and the displacement will be performed; if the distance is considerable, the AI character may have already dodged when the distance was exceeded, therefore, the dodge action and movement are stopped at this time. The construction of Example 1 has been completed.

5.2. Example 2

Since many well conFig. parts can be used from Example 1, Example 2 will be simpler and faster to implement, and reuse is one of the benefits of modularity. The description of the settings that are will be omitted.

5.3. Set the Main Body of the Animation Game

The main function of the animation game is roughly the same as in Example 1. The difference is that in Example 2, a new spear class, inherited from BasicModel, is added. Spear is used to represent the spear thrown by the AI character, and the spear needs to be independently updated after being thrown. Therefore, a basic model series, relating to the spear, is set up in the ModelManager, and the spear is also added to the Update and Draw functions: operation and establishment of AI system-AI manager and FuSM. In the AI system during the implementation of Example 2, the aim was to highlight the modular reusability and design flexibility, therefore, during the design process, part of the architecture of Example 1 was combined with the new architecture of Example 2. The aim in the second example was to implement an AI role that was significantly different from the first example. Therefore, the second example selected the implementation of a long-range, attacking AI role. Table 7 below shows the behaviour list of the second AI role.

The AI character in Example 2 has four behaviour modes: patrol, chasing the player, combat movement and long-range attack. The two behaviours of patrol and chasing the player are the examples shown in Example 1, on the one hand, to demonstrate the reusability and on the other hand, because the two behaviours of patrolling and chasing players are almost necessary behaviours for AI characters in action animation games. Similar to Example 1, the four behaviours are divided into the combat element and the non-combat element. The non-combat element also uses FSM architecture to establish the two behaviours of patrol and chasing the player, but the difference is the use of the combat element. The FuSM is used to establish the two behaviours of combat movement and long-range attack. Fig. 8 is the AI system planning of Example 2.

1. EnemyAIManager: the EnemyTwoAIManager is created in the EnemyTwo category, as the AI manager of Example 2. The part of the EnemyTwoAIManager setting that connects the limited state machine is the same as in Example 1. The two states of EnemyPatrolState and EnemyChaseState are the objects created directly using Example

1. Only the combat state part is an additional EnemyFightTwoState state. The information processed by EnemyTwoAIManager is different from that in Example 1. The information here is used to determine whether the fuzzy state is enabled. The two states in the FuSM are based on the distance between the AI character and the player, so as to judge whether or not to enable, therefore, this parameter must be managed in EnemyTwoAIManager, in order to make the FuSM available for use.

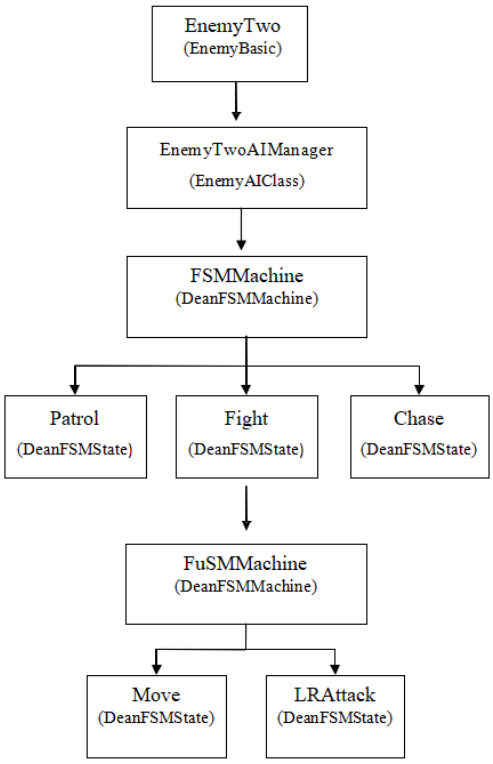


Fig. 8. Example 2 AI system planning

The following describes the two states of EnemyFuMoveState and EnemyFuAttackState, respectively.

1. EnemyFuMoveState: EnemyFuMoveState is the state of controlling the movement of combat. There are three behaviours. The first is to stand in one place, the second is to go around the centre of the circle with the player and the third is to move backwards when the player approaches, in an attempt to keep a distance from the player.

2. EnemyFuAttackState: the function of the FuSM is to allow the two states of EnemyFuMoveState and EnemyFuAttackState to operate independently, allowing the AI character to attack the player at a distance while moving. The behaviour of the EnemyFuAttackState is very simple and involves throwing a spear at the player at a fixed time.

Combining Two Implementation Examples

Example 2 shortens the implementation time considerably due to the modularity and quickly adds a new AI role by expanding Example 1. Then, the two examples are combined. Example 2 used many objects of Example 1 in its implementation and adds a small part of the program while maintaining the object structure, so Example 2 is an extension of Example 1. Therefore, we take Example 2 as the main body, and integrate a part of Example 1 in Example 2. Firstly, it is necessary to identify the objects that are different in Example 1 and Example 2, then the objects of both are gathered together. Following a comparison, the objects in Example 1, that are different from Example 2, are EnemyOne, EnemyOneAIManager and EnemyFightState. These three objects are added to the project in Example 2. The next step is to start to connect the program parts of the two examples. Firstly, the combination of the two examples are examined in the form of a framework diagram. As shown in Fig. 9, there is no conflict between the two examples and the only connecting part is EnemyOne. There is a link between EnemyTwo and ModelManager, therefore the only change required is to add a new AI role of EnemyOne to the EnemyModels list in ModelManager and the implementation of Example 2 has been completed.

Performance Evaluation and Discussion

This study used the XNA Game Studio, which simplifies the development of animation game programs in order to implement the AI system of action animation games, combined with AI technologies such as FSMs, FuSMs and neural networks, so that AI technology can be used. The results of this paper, namely, the concept of modular objects, allows the AI system to exhibit features such as scalability, reusability and design flexibility. The implementation process records and experience of this research can provide useful information for related research reference for both the creator and the producer. The research conclusions and contributions of this paper are as follows:

1. Achievements: an AI animation game system with reusability, design flexibility and expandability has been built, using the concept of object-oriented modular objects. As a result of the use of FSMs, FuSMs and neural networks, AI technology applications are added to combine traditional rule-based systems and learning adaptation systems, so as to increase the learning ability for traditional AI roles.

2. Two examples of AI roles with different behaviour patterns are implemented: the AI role behaviour in the first example is based on close-range attacks. The AI system architecture uses a combination of FSMs and neural networks and uses traditional rule-based methods. In Example 1, the implementation process from scratch verifies the feasibility of the AI system. Example 2 is completed practically through the combination of various built objects and through the combination of FSMs and neural networks. AI performance also achieved the expected results.

3. The AI character behaviour of the second example is aimed to be a long-range attack type and the combination of an FSM and an FuSM is used in the architecture of the AI system. The FSM element is built according to the first example. Fuzzy logic is used to allow the AI character to change its movement speed and movement method as the player approaches. The behaviour of AI characters is designed to have more

variability, and Example 1 is combined with Example 2 by means of simple steps, so that the two AI characters can be presented in the animation game at the same time. During the implementation of Example 2, many objects created in Example 1 were reused. This verifies the reusability and expandability in this research. These two features allow animation game designers to continuously reuse previously built objects and can also expand the functions of the objects to create an animation game design.

4. When an animation game designer tries out different AI technologies, he/she simply needs to expand or modify the original objects, without the need for a complete redesign. The difference between the two examples in the AI system architecture aims to highlight the design flexibility achieved by this research study. Since each AI system can operate independently, the animation game designers can integrate different AI technologies according to their needs and all AI systems can operate within animation games at the same time without affecting one other. Therefore, this research has reached the expected goals in terms of the implementation results.

5. XNA simplifies the effectiveness of animation game programming: XNA can simplify animation game design during the process of implementing the AI system and can explain and record the implementation process, so as to provide a reference for related researchers and producers.

The previously created animation game module objects are used to gradually build a prototype animation game, and the FSM and neural network technology is used to create an AI character that combines fixed and variable behaviours, as demonstrated in Example 1. The feasibility of the AI system and the combined effect of modular objects are discussed. During the implementation of Example 2, another AI character was quickly created by following the system established in Example 1 and completed using a different combination of AI technologies from Example 1, which demonstrates the reusability of the AI system, as well as design flexibility. The combination of the two examples shows the independence of the AI system. Combining the above characteristics can allow the AI system of this study to quickly create diversified AI roles, enabling each AI role to have sufficient design flexibility, which is not limited to a fixed architecture.

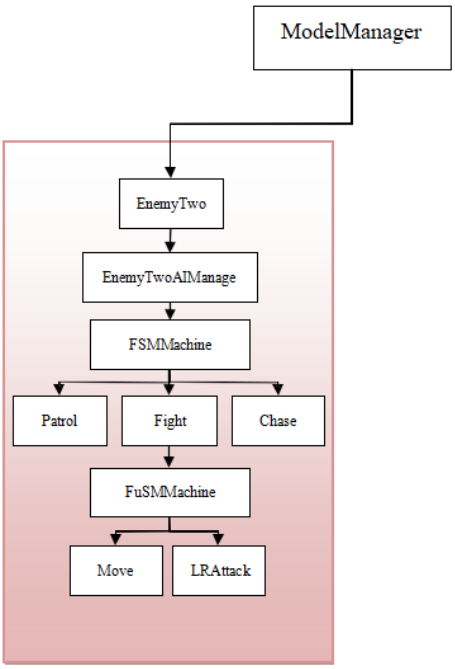


Fig. 9 Architecture diagram after the two examples are combined

6. Conclusion and Future work

This study used XNA Game Studio, which simplifies the development of animation game programs, so as to implement the AI system of action animation games, combined with AI technologies such as FSMs, FuSMs and neural networks, making use of AI technology. The results of this paper demonstrate the concept of modular objects, which allow the AI system to exhibit features such as scalability, reusability and design flexibility. Through the implementation process, records and experience of this research can provide useful information for related research, both for the creator and the producer. The research conclusions and contributions of this paper are as follows:

1. Achievements: an AI animation game system has been built with reusability, design flexibility and expandability, using the concept of object-oriented modular objects. By making use of FSMs, FuSMs and neural networks, AI technology applications are added to combine traditional rule-based systems and learning adaptation systems to increase the learning ability of traditional AI roles.

2. Two examples of AI roles with different behaviour patterns are implemented: the AI role behaviour in the first example is based on close-range attacks and the AI system architecture uses a combination of FSMs and neural networks, as well as traditional rule-based methods. In Example 1, the implementation process from scratch verifies the feasibility of the AI system and Example 2 is a practical example, created from a

combination of various built objects and the combination of FSMs and neural networks. AI performance has also achieved the expected results.

3. The AI character behaviour of the Example 2 is planned to be a long-range attack type, and the combination of an FSM and an FuSM is used in the architecture of the AI system. The FSM element is built according to the first example. Fuzzy logic is used to allow the AI character to change its movement speed and movement method as the player approaches. The behaviour of AI characters is given more variability and Example 1 and Example 2 are combined through simple steps, so that the two AI characters can be presented in the animation game at the same time. During the implementation of Example 2, many objects created in Example 1 were reused, which verifies the reusability and expandability in this research. These two features allow animation game designers to continuously reuse previously built objects and also to expand the functions of the objects to create an animation game design.

4. When an animation game designer aims to try out different AI technologies, he/she only needs to expand or modify the original objects without the need for a complete redesign. The difference between the two examples in the AI system architecture is to verify the design flexibility that this research expects to show. Since each AI system can operate independently, animation game designers can integrate different AI technologies according to their needs, and all AI systems can operate within animation games at the same time, without affecting one other. Therefore, this research has reached the expected research goals in the implementation results.

5. XNA simplifies the effectiveness of animation game programming: XNA can simplify animation game design by implementing the AI system and explaining and recording the implementation process to provide a reference for related researchers and producers.

The following are limitations of the AI system in this research study. 1. The lack of information hiding: the purpose of information hiding is to prevent the important information inside the object from being arbitrarily changed, since changing the information may damage the internal operating structure of the object. 2. The lack of a scripted design: at present, the parameter settings of the AI system must be directly modified in the program. When the animation game is large in scale, this will be costly in terms of compiling the project time. Future work: future research work will focus on the visual design of the system, as the programmers only provide technical support. The actual AI animation game is usually designed by the planners. If the planners in general have no knowledge of programming, it is relatively important to build an AI system that they can use. Therefore, creating a visual editor for the AI system will be one of the focuses of future research.

References

1. Habibie, I., Holden, D., Schwarz, J., Yearsley, J., Komura, T.: A recurrent variational autoencoder for human motion synthesis, Proc. 28th Brit. Vis. Conf. (BMVC), (2017).
2. Harvey, F., Pal, C.: Semi-supervised learning with encoder-decoder recurrent neural networks: Experiments with motion capture sequences, Comput. Sci., 3, 553-562 (2015).
3. Holden, D., Habibie, I., Kusajima, I., Komura, T.: Fast neural style transfer for motion data, IEEE Comput. Graph. Appl., 37(4), 42-49 (2017).

4. Holden, D., Saito, J., Komura, T.: A deep learning framework for character motion synthesis and editing, ACM Trans. Graph., 35(4), 138. (2016).
5. Kwon, T., Shin, S.: Motion modeling for on-line locomotion synthesis, Proc. ACM SIGGRAPH/Eurograph. Symp. Comput. Animation, 29-38 (2005).
6. Makuch, E.: Tomb Raider 1-5 hit Steam". GameSpot. Archived from the original in 2019. Retrieved 14 (2020).
7. Mandery, C., Terlemez, O., Do, M., Vahrenkamp, N., Asfour, T.: Unifying representations and large-scale whole-body motion databases for studying human motion, IEEE Trans. Robot., 32(4), 796-809 (2016).
8. Martinez, J.: Black, M., Romero, J.: On human motion prediction using recurrent neural networks, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2891-2900 (2017).
9. Mehta D, et al.: Monocular 3D human pose estimation in the wild using improved CNN supervision, Proc. Int. Conf. 3D Vis., pp. 506-516. (2017)
10. Merel J, et al.: Learning human behaviors from motion capture by adversarial imitation, Available: https://arxiv.org/abs/1707.02201. (2017)
11. Roberts, S.: Age of Empires 4 is 'making good progress', and Microsoft will talk about it later this year". PC Gamer. Archived from the original in June, 2019. Retrieved 12 June, (2019).
12. Sato: Final Fantasy XIV Reaches 14 Million Adventurers Worldwide. Siliconera. Retrieved (2018).
13. Ke X, Zou J, Niu Y. End-to-End Automatic Image Annotation Based on Deep CNN and Multi-Label Data Augmentation. IEEE Transactions on Multimedia, 21(8): 2093-2106. (2019)
14. Shin S, Kim C: Human-like motion generation and control for humanoid's dual arm object manipulation, IEEE Trans. Ind. Electron., vol. 62, no. 4, 2265-2276. (2015)
15. Skrebels J.: Super Mario 3D World + Bowser's Fury Announced for Nintendo Switch. IGN. Archived from the original in September, Retrieved 3 September. (2020)
16. Tan Q, Gao L, Lai Y, Yang J, Xia, S.: Mesh-based autoencoders for localized deformation component analysis, Proc. 30th AAAI Conf. Artif. Intell., 1-8. (2018)
17. Tom, M.: XNA Game Studio 4.0 Programming: Developing for Windows Phone 7 and Xbox 360. Pearson Education. ISBN 9780132620130. (2010)
18. Cheng, H., Wu, L., Li, R., et al.: Data recovery in wireless sensor networks based on attribute correlation and extremely randomized trees. Journal of Ambient Intelligence and Humanized Computing, 12(1): 245-259. (2021)
19. Cheng, Y., Jiang, H., Wang, F., et al.: Using High-Bandwidth Networks Efficiently for Fast Graph Computation. IEEE Transactions on Parallel and Distributed Systems, 30(5): 1170-1183. (2019)
20. Dai, Y., Wang, S., Chen, X., et al.: Generative adversarial networks based on Wasserstein distance for knowledge graph embeddings. Knowledge-Based Systems, 190: 105165. (2020)
21. Feng Y, et al.: Mining spatial-temporal patterns and structural sparsity for human motion data denoising, IEEE Trans. Cybern., vol. 45, no. 12, 2693-2706. (2015)
22. Fu, Y. G., Huang, H. Y., Guan. Y., et al.: EBRB cascade classifier for imbalanced data via rule weight updating. Knowledge-Based Systems, 223: 107010. (2021b)
23. Fu, Y. G., Ye, J. F., Yin, Z. F., et al.: Construction of EBRB classifier for imbalanced data based on Fuzzy C-Means clustering. Knowledge-Based Systems, 234: 107590. (2021a)
24. Fu, Y. G., Zhuang, J. H,, Chen. Y, P., et al.: A framework for optimizing extended belief rule base systems with improved Ball trees. Knowledge-Based Systems, 210: 106484. (2020)
25. Li Z, Zhou Y, Xiao S, He C, Huang Z, Li, H.: Auto-conditioned recurrent networks for extended complex human motion synthesis, Available: https://arxiv.org/abs/1707.05363. (2017)

26. Li, X. Y., Lin, W., Liu, X., et al.: Completely Independent Spanning Trees on BCCC Data Center Networks with an Application to Fault-Tolerant Routing. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 33(8): 1939-1952. (2022)
27. Liu, G., Chen, X., Zhou, R., et al.: Social learning discrete Particle Swarm Optimization based two-stage X-routing for IC design under Intelligent Edge Computing architecture. Applied Soft Computing. 10, 107215. (2021)
28. Liu, G., Chen, Z., Zhuan,g Z., et al. A unified algorithm based on HTS and self-adapting PSO for the construction of octagonal and rectilinear SMT. Soft Computing, 24(6): 3943-3961. (2020a)
29. Liu, G., Zhang, X., Guo, W., et al.: Timing-Aware Layer Assignment for Advanced Process Technologies Considering Via Pillars. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 41(6): 1957-1970. (2022a)
30. Liu, G., Zhu, Y., Xu, S., et al.: PSO-Based Power-Driven X-Routing Algorithm in Semiconductor Design for Predictive Intelligence of IoT Applications. Applied Soft Computing, 114: 108114. (2022b)
31. Liu, N., Pan, J. Sun, C., et al.: An efficient surrogate-assisted quasi-affine transformation evolutionary algorithm for expensive optimization problems. Knowledge-Based Systems, 209: 106418. (2020c)
32. Lu, Z., Liu, G., and Wang, S.: Sparse neighbor constrained co-clustering via category consistency learning. Knowledge-Based Systems, 201, 105987. (2020)
33. Wang, S., Wang, Z., Lim, K. L., et al.: Seeded random walk for multi-view semi-supervised classification. Knowledge-Based Systems, 222: 107016. (2021)
34. Xia, S., Wang, C., Chai, J., Hodgins, J.: Realtime style transfer for unlabeled heterogeneous human motion, ACM Trans. Graph., vol. 34, 119.. (2015)
35. Yu, Z, Zheng, X, Huang, F, et al.: A framework based on sparse representation model for time series prediction in smart city[J]. Frontiers of Computer Science, 15(1): 1-13. (2021)
36. Yumer, M., Mitra, N.: Spectral style transfer for human motion between independent actions, ACM Trans. Graph., vol. 35, no. 4, 137. (2016)
37. Zhang, H., Li, J. L,, Liu, X. M., et al.: Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection[J]. Future Generation Computer Systems, 122: 130-143. (2021a)
38. Zhang, Y., Lu, Z., and Wang, S.: Unsupervised feature selection via transformed auto-encoder[J]. Knowledge-Based Systems, 215: 106748. (2021b)
39. Zheng, X., Rong, C, et al.: Foreword to the special issue of green cloud computing: Methodology and practice[J]. Concurrency and Computation: Practice and Experience. 31(23): e5425. (2019)
40. Zhou, X., Zhu, M., Pavlakos, G., Leonardos, S., Derpanis, K., Daniilidis, K.: MonoCap: Monocular human motion capture using a CNN coupled with a geometric prior, IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 4, 901-914. (2019)
41. Zou, W., Guo, L., Huang, P., et al.: Linear time algorithm for computing min-max movement of sink-based mobile sensors for line barrier coverage. Concurrency and Computation: Practice and Experience, 34(2): e6175. (2022)
42. Zhong, S, Jia C, Chen K, et al.: A novel steganalysis method with deep learning for different texture complexity images. Multimedia Tools and Applications, 78(7): 8017-8039. (2019)
43. Pu L, Zhu D, Jiang H.: A 1.375-approximation algorithm for unsigned translocation sorting. Journal of Computer and System Sciences, 113: 163-178. (2020)
44. Shen, S., Yang, Y., Liu, X.: Toward data privacy preservation with ciphertext update and key rotation for IoT. Concurrency and Computation: Practice and Experience, e6729. (2021)
45. Pan, W., Zhao, Z., Huang, W., Zhang, Z., Fu, L., Pan, Z., Yu, J., Wu, F.: Video moment

retrieval with noisy labels. IEEE Transactions on Neural Networks and Learning Systems, doi: 10.1109/TNNLS.2022.3212900. (2022)
46. Ma, L., Zheng, Y., Zhang, Z., Yao, Y., Fan, X. Ye, Q.: Motion Stimulation for Compositional Action Recognition, IEEE Transactions on Circuits Systems and Video Technology, 2022, Early Access. (2022)
47. Fu, L., Zhang, D. and Ye, Q.: Recurrent Thrifty Attention Network for Remote Sensing Scene Recognition, IEEE Transactions on Geoscience and Remote Sensing, vol.59, no.10, 8257-8268. (2021)
48. Ye, Q., Huang, P., Zhang, Z., et al.,: Multi-view Learning with Robust Double-sided Twin SVM with Applications to Image Recognition, IEEE Transactions on Cybernetics, vol.52, no.12, 12745 - 12758. (2022)
49. Fu, L., Li, Z., Ye, Q., et al.: Learning Robust Discriminant Subspace Based on Joint L2,p- and L2,s-Norm Distance Metrics, IEEE Transactions on Neural Networks and Learning Systems, vol.33, no.1,130 -144. (2022)
50. Chen, X., Li, M., Zhong, H., Ma, Y. and Hsu, C.: DNNOff: Offloading DNN-based Intelligent IoT Applications in Mobile Edge Computing. IEEE Transactions on Industrial Informatics, 18(4): 2820-2829. (2022a)
51. Chen, X., Zhang, J., Lin, B., Chen, Z.: Katinka Wolter, Geyong Min. Energy-Efficient Offloading for DNN-based Smart IoT Systems in Cloud-Edge Environments. IEEE Transactions on Parallel and Distributed Systems, 33(3): 683-697. (2022b)
52. Chen, X., Hu, J., Chen, Z. Lin, B., Xiong, N., Min, G..: A Reinforcement Learning Empowered Feedback Control System for Industrial Internet of Things. IEEE Transactions on Industrial Informatics, 18(4): 2724-2733. (2022c)
53. Chen, X., Yang, L., Chen, Z., Min, G., Zheng, X. Rong, C.: Resource Allocation with Workload-Time Windows for Cloud-based Software Services: A Deep Reinforcement Learning Approach. IEEE Transactions on Cloud Computing, Publish Online, DOI: 10.1109/TCC.2022.3169157. (2022d)
54. Huang, G., Luo, C., Wu, K., Ma, Y., Zhang, Y, and Liu, X.: Software-Defined Infrastructure for Decentralized Data Lifecycle Governance: Principled Design and Open Challenges. IEEE International Conference on Distributed Computing Systems. (2019)

Rong Zhang received a bachelor's degree from Wuhan Institute of Technology, China, in 2006, and MS degree from Hubei University of Technology, Wuhan, China, in 2011. She is currently a lecturer with the School of Artificial Intelligence, Dongguan Polytechnic. Her major research interests include Artificial Intelligence, Internet of Things and animation game.

Received: July 19, 2022; Accepted: November 11, 2022.

Computer Science and Information Systems 20(2):793–812 https://doi.org/10.2298/CSIS220815016F

Pedestrian Attribute Recognition Based on Dual Self-attention Mechanism

Zhongkui Fan[1] and Ye-peng Guan[1, 2]

[1] School of Communication and Information Engineering, Shanghai University, 200444 Shanghai, China
{fanzkui, ypguan}@shu.edu.cn
[2] Key Laboratory of Advanced Displays and System Application, Ministry of Education, 200444 shanghai, China

Abstract. Recognizing pedestrian attributes has recently obtained increasing attention due to its great potential in person re-identification, recommendation system, and other applications. Existing methods have achieved good results, but these methods do not fully utilize region information and the correlation between attributes. This paper aims at proposing a robust pedestrian attribute recognition framework. Specifically, we first propose an end-to-end framework for attribute recognition. Secondly, spatial and semantic self-attention mechanism is used for key points localization and bounding boxes generation. Finally, a hierarchical recognition strategy is proposed, the whole region is used for the global attribute recognition, and the relevant regions are used for the local attribute recognition. Experimental results on two pedestrian attribute datasets PETA and RAP show that the mean recognition accuracy reaches 84.63% and 82.70%. The heatmap analysis shows that our method can effectively improve the spatial and the semantic correlation between attributes. Compared with existing methods, it can achieve better recognition effect.

Keywords: pedestrian attribute recognition; spatial self-attention; semantic self-attention; deep learning.

1. Introduction

Visual recognition of pedestrian attributes has recently drawn a large amount of research attention due to its great potential applications in computer vision. Pedestrian attributes are defined as semantic mid-level descriptions of people, such as gender, age, hairstyle, body fat or thin, clothing style, and accessories. The research has achieved much success in many fields such as image retrieval [1,2], object recognition [3,4], and person re-identification [5]. It has also shown great application prospects in smart video surveillance and video-based business intelligence. At present, pedestrian attributes recognition is a great challenge, the main difficulties are: (1) poor image quality, low resolution, occlusion, motion blur, etc.; (2) uncontrollable interference, such as illumination and camera viewing angle, which aggravate the difficulties of feature recognition. (3) some pedestrian attribute recognition tasks require local fine-grained information, such as “glasses”.

There are some pioneering works on attribute recognition in multiple application scenarios. Layne et al. [6] first using Support Vector Model(SVM) to recognize attributes (e.g. “gender”, “backpack”) to assist pedestrian re-identification. To solve the attribute recognition problem in mixed scenarios, Zhu et al. [7] use boosting algorithm to recognize attributes. Deng et al. [8] construct the pedestrian attribute database (PETA) and utilize SVM and Markov Random Field to recognize attributes. However, these methods [9,10,11] use handcraft features, which cannot recognize the attributes effectively. Due to the outstanding performance of CNN in computer vision tasks, some researchers have tried to use it for pedestrian attribute recognition. [12] proposed CNN-based methods DeepSAR and DeepMAR for pedestrian attribute recognition to achieve better results. [13] proposed segment the pedestrian into multiple parts, and then recognize the attributes of each part. However, these simple methods cannot accurately segment pedestrians and require pre-processing of images. In addition, the correlation among attributes is ignored, which is very important to attributes recognition. For example, longhair feature has a higher probability for women than men, so the hair length could help to recognize the gender. Siddiquie et al. Zhang et al. [14] propose a pose aligned neural networks to recognize pedestrian attributes (e.g. age, gender) on images in multiple scenarios. The method poselets[15] decompose the image into a set of parts, which provide a robust distributed representation of a person, attributes can be inferred without explicitly localizing body parts. Although Pedestrian attributes recognition has been studied for years, it is still a quite challenge in real application environments. Pedestrian attribute recognition includes both local fine-grained and global feature recognition. In low-resolution images, some local attributes (such as glasses) occupy a small area, and the attributes need to be identified through fine-grained features, some abstract attributes (such as gender) need to be judged by overall features. At the same time, some attributes are spatial correlation, and some attributes are semantical correlation (such as gender attributes and skirt attributes), it can be used to improve the recognition effect between attributes.

To overcome the shortcomings of current pedestrian attribute recognition, this paper designs a dual self-attention convolution neural network (CNN) is proposed. In the work, the spatial and semantic self-attention are used for locating key regions, and the key regions are paid more attention to pedestrian attributes recognized. Secondly, using the relationship between attributes to improve the recognition accuracy. The proposed methods obtain state-of-the-art results on datasets PETA and RAP.

In this paper, there are three contributions.

(1) An end-to-end pedestrian attribute recognition framework is proposed. feature extraction, key points location, and pedestrian attributes recognition form an end-to-end network.

(2) The dual attention mechanism is proposed. Spatial self-attention is used for finding out pedestrian attributes from regions, and semantic self-attention is used to obtain the constraints among attributes.

(3) A hierarchical recognition strategy is proposed. We classify the pedestrian attributes as global (such as gender and age) and local (such as hairstyle and has glass), the whole regions are used for the global attribute recognition, and the partial regions are used for the local attribute recognition.

The organization of the rest of this paper is as follows. In Section II, we introduce the research and development of pedestrian attribute recognition. In Section III, the pedestrian attribute recognition framework is introduced in particular. In this section,

we divide the methods into several parts and discuss detailed process in each part respectively. The experiments are carried on the benchmark dataset and the experimental results are analysed in Section IV. In Section VI, we give a conclusion.

2. Related Work

Pedestrian attribute recognition is important to research in computer vision. Given a person's image, pedestrian attribute recognition aims to predict a group of attributes to describe the characteristics of the person from a pre-defined attribute list. In the past decade, researchers have done a lot of research on pedestrian attribute recognition, and many methods have achieved good results, which can be divided into two groups: hand-crafted features and deep learning-based methods.

2.1. Pedestrian Attribute Recognition with Hand-Crafted Features

Pedestrian attributes were first used for human recognition by [16], the person image is divided into multiple regions, and each region is associated with a classifier based on Haar-like features. Then, the attribute information is used to retrieve surveillance video streams. The approach proposed by [10] extracts a 2784 dimensional low-level color and texture feature vector for each image and trains an SVM for each attribute. The attributes are further used as a mid-level representation to help person re-identification. [17] introduced the pedestrian attribute database APiS. Their method determines the upper and lower body regions according to the average image and extracts color and gradient histogram features (HSV, MB-LBP, HOG) in these two regions, then, an Adaboost classifier is trained to recognize attributes. The drawback of these approaches is that accuracy is low and the correlation between attributes is not considered. To overcome this limitation, [18] proposed an interaction model, based on Adaboost approach, learning an attribute interaction regressor. The final prediction is a weighted combination of the independent score and the interaction score. [19] constructed the pedestrian attribute dataset "PETA", which uses a Markov Random Field (MRF) to predict the relation between attributes, attributes are recognized by exploiting the context of neighbouring images on the MRF-based graph. [20] uses a multilabel Multi-layer perceptron to classify all attributes at the same time.

2.2. Pedestrian Attribute Recognition with Deep Learning

In recent years, deep learning has achieved success in automatic feature extraction using multi-layer nonlinear transformation, methods based on Convolutional Neural Network (CNN) models have been proposed for pedestrian recognition.[12] fine-tuned the CaffeNet trained on ImageNet to perform single and multiple attribute recognition. Sudowe et al. [21] proposed ACN(Attributes Convolutional Net), which uses an Alexnet network to extract features and set a classifier for each attribute to realize attribute recognition. Abdulnabi[22] proposed Multi-Task Convolutional Neural Network (MTCNN) , which shares feature pools each task corresponds to an attribute

recognition. The above methods only extract the overall features of pedestrians, and pedestrian attribute recognition can be regarded as a fine-grained multi-label classification task. Highlighting local attribute features can greatly improve the recognition effect. To enhance the significance of local features, the existing methods extract local features by segmenting pedestrian images. For example, [18] proposed to divide the pedestrian images into 15 overlapping parts where each part connects to several CNN pipelines with several convolution and pooling layers. Some methods extract the local features of trunk position and use the correlation between local features for attribute recognition. In method [23], the pedestrian image is horizontally cut into multiple regions, and the features of each region are encoded and decoded, to make the local features more prominent. Another way is to use the attention mechanism to improve the accuracy of attribute recognition by assigning feature weights. Typical methods include the Spatial Regularization Network (SRN)[24], which applies the attention mechanism to the pedestrian attribute recognition for the first time, the Hydraplus-Net [25] proposes a multi-directional attention mechanism, and JVRKD(Join Visual-semantic Reasoning and Knowledge Distillation) [26], which combines local feature methods and attention mechanism methods.

2.3. Pedestrian Attribute Recognition with self-attention

In convolutional networks for image recognition, the layers of the network perform two functions. The first is feature aggregation, which the convolution operation performs by combining features from all locations tapped by the kernel. The second function is feature transformation, which is performed by successive linear mappings and nonlinear scalar functions: these successive mappings and nonlinear operations shatter the feature space and give rise to complex piecewise mappings. In computer vision tasks, the correlation between features can effectively improve accuracy. In order to obtain features outside the local area, it is necessary to increase the receptive field of deep neurons through the stacking of convolutional layers, which will increase the complexity of the convolutional neural network. Inspired by the idea of a No-local mean de-noising filtering algorithm [27], Wang et al. [28] proposed the No-Local network for video classification. In the literature [29], combining Transformer and No-Local, a self-attention mechanism was proposed to solve the problem of non-local feature dependence. In the task of image classification and pedestrian re-recognition, the literature [30, 31] used the self-attention mechanism to achieve better results. However, the method in the [27- 30] only uses the spatial correlation of features and does not use the correlation between channels. Literature [31] proves that semantic information has a strong correlation with channel features. [32-34] combine spatial attention and channel attention for image classification.

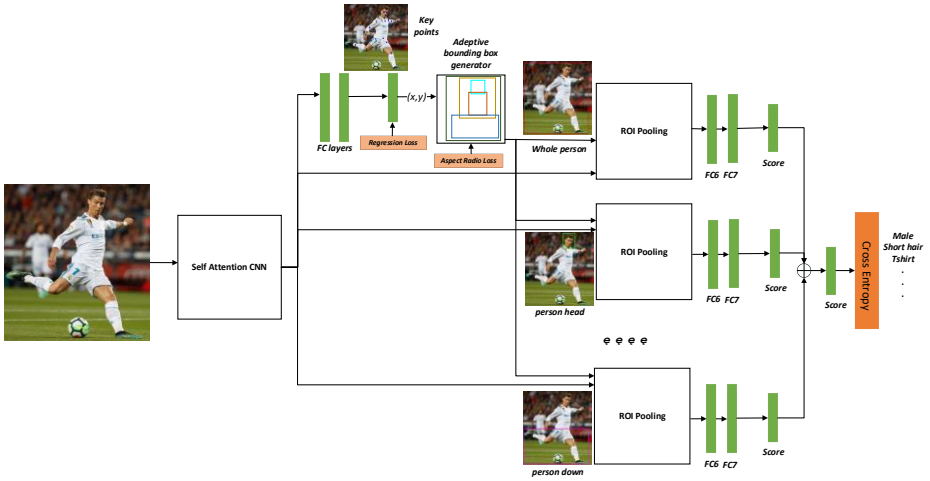


Fig.1. The pipeline of dual self-attention pedestrian attributes recognition. It consists of initial convolutional feature exaction layers, a key point localization network, an adaptive bounding box generator for each part, and the final attribute classification network for each part.

3. Methods

This paper proposes an end-to-end framework for pedestrian attributes recognition. In the method, high-quality, unique, and distinguishable features are obtained through spatial and semantic self-attention mechanisms, which are used for the pedestrian key points (Such as left eye, right eye, nose, left shoulder, right shoulder, etc. 17 points) are estimated, from the key points, the bounding boxes are generated adaptively. Instead of applying the transform to the whole image, we apply a spatial transform for each part and use the bilinear sampler to get the image features for subsequent attribute recognition. The whole network is learned end-to-end in a multi-task setting, attribute recognition as the main task and person key point prediction as to the auxiliary one. the framework is exemplified in Figure 1.

3.1. Dual self-Attention CNN

Good features are the premise for pedestrian attribute recognition. In order to obtain high-quality features, this paper proposes the dual attention mechanism of spatial self-attention and semantic self-attention. The spatial self-attention mechanism is used to salience the features that have correlation in space, and the semantic self-attention mechanism is used to salience the features that have correlations in channels. The network structure is shown in Figure 2.

The dual attention network is divided into four stages. From the top to the bottom of each subnet, the resolution of each level is gradually reduced by a multiple of 1/2, and the number of channels is increased by a factor of 2. The first stage consists of a high-resolution subnet, which contains 4 linearly transformed E-ecaneck residual modules.

The second, third, and fourth stages are composed of an improved E-ecablock module and a fusion module (Non-local block) of the spatial attention mechanism. The goal is to extract and merge features more extensively and deeper. The E-ecaneck module and E-ecablock module are constructed by channel attention as the basic modules to maximize the acquisition of useful channel information, Then integrate the Non-local block module in the spatial attention mechanism based on the channel attention mechanism to realize the effective extraction and fusion of spatial information. Finally, through the up-sampling operation, the features are output to realize the task of detecting key points of the human body and further realize the estimation of the human body posture.

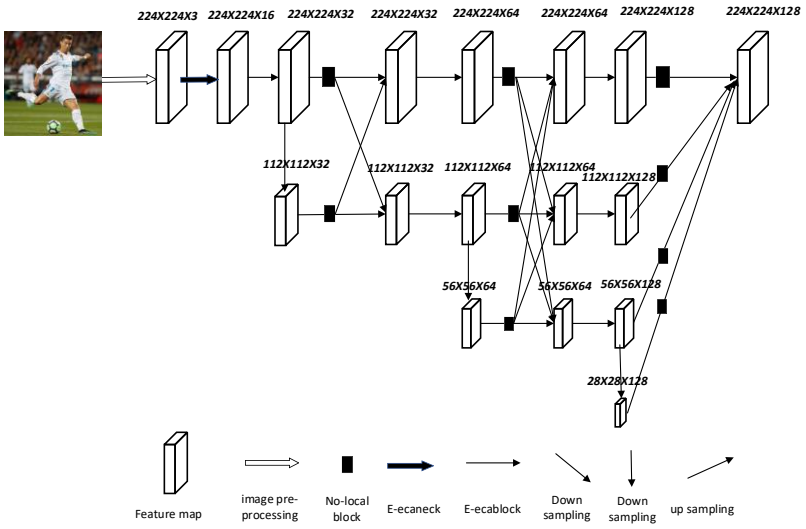


Fig 2. self-attention CNN. Features are extracted from the spatial self-attention network and semantic self-attention network.

Spatial Self-attention non-Local block

In deep learning networks, the dependency between features at different locations plays an important role in computer vision task. No-local has the following characteristics, Firstly, which calculates the features of a certain location as the weighted sum of the features of other locations; Secondly, which directly captures dependencies by calculating the interaction between any two locations, regardless of their distance; Third, which unfixed input size and can be embedded in many deep learning network structures. The non-local block structure is shown in Figure 3.

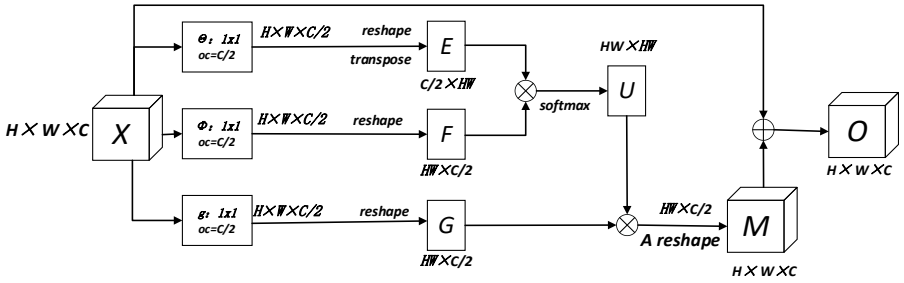


Fig.3. Non-local block

The Non-local block module is the core module of the spatial domain attention model (Non-local Neural Networks). As a non-local spatial attention method, it is not only limited to the local area but operates in the global area. The calculation formula is as follows the equation (1) shows:

$$y_i = \frac{1}{c(x)} \sum_{\forall j} f(x_i, x_j) g(x_j) \tag{1}$$

In equation (1), i and j are some spatial bits, $c(x)$ is the normalization factor, and x and y are input and output data with the same size respectively. x_i is a vector, $f(x_i,x_j)$ is to calculate the correlation function between x_i and all x_j. $g(x_j)$ calculate the characteristic value of the input signal at position j. After formula (1), formula (2) can be obtained, where Z_i is the final output, which is the linear conversion matrix realized by the 1×1 convolution operation.

$$z_i = w_z y_i + x_i \tag{2}$$

In this paper, the spatial attention module (non-local block) module is used to improve the fusion of the second, third and fourth stages, and it is added to the fusion of each resolution characterization based on integrating the channel attention, as shown in Figure 4. Shown. The non-local block module is operated in the global region, which can expand the receptive field. Therefore, it can effectively extract more beneficial information by different resolutions used for multi-scale fusion to get a better effect.

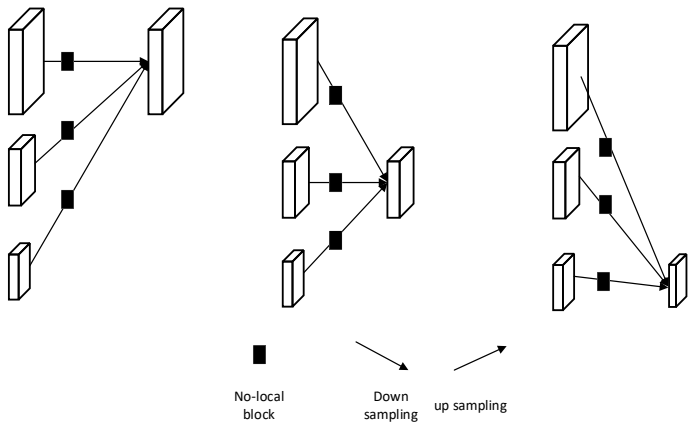


Fig.4. Integration spatial attention Integration module

Efficient Channel Attention network

The ECA module is a method of capturing local cross-channel information interaction. It conducts cross-channel information interaction without reducing the channel dimension and aims to ensure computational performance and model complexity. The ECA module can realize information interaction between channels through one-dimensional convolution with a convolution kernel size *K*, as shown in Figure 5.

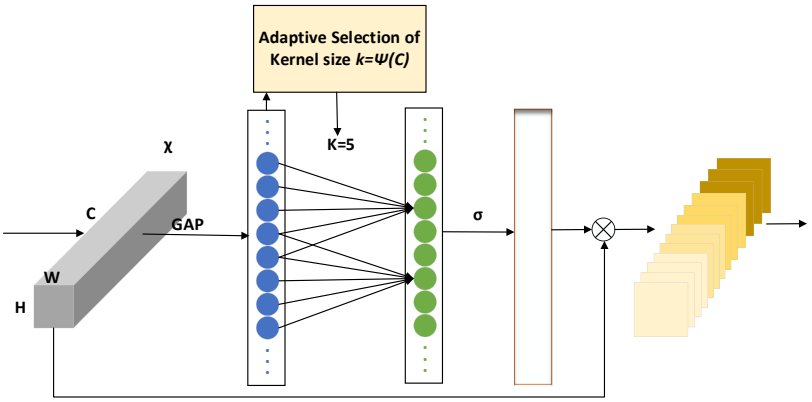


Fig.5. ECA (Efficient Channel Attention network) structure diagram

The output of a convolution block is $\chi \epsilon R_{H\times W\times C}$, *W*, *H*, *C*, represent width, height, and number of channels, respectively, and GAP represents global average pooling. The ECA module passes a one-dimensional convolution with a convolution kernel size of *k*. To realize the information exchange between channels, as shown in formula (1) Where σ is a *Sigmoid* function, *C1D* represents one-dimensional convolution, and *k* represents *k* parameter information, which is the output signal. This method of capturing cross-channel information interaction ensures performance results and model efficiency.

$$\omega = \sigma\left(C1D_K(y)\right) \tag{3}$$

Inspiring from the method of bottleneck and basic block in ResNet network, this paper integrates ECABlock into the bottleneck (bottleneck) and residual (basic block) to get an improved E-ecablock and E-ecaneck, which replace the bottleneck module and basic block module of the original network, to realize cross-channel information interaction without reducing the channel dimension.

E-ecablock module and E-ecaneck module

The E-ecablock module proposed in this paper contains two ECABlock modules with the size of the 3×3 convolution kernel and one residual connection, as shown in Figure 6. The E-ecaneck module includes two ECABlock modules with a size of 1×1 convolution kernel, one ECABlock module with a size of 3×3 convolution kernel, and a residual connection, as shown in Figure 7.

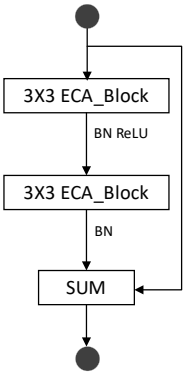


Fig.6. E-ecablock

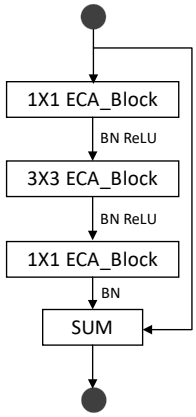


Fig.7. E-ecaneck

Inspired by the literature [35], this article adds the channel attention module before the convolution to get better results in feature extraction. This article adds two 3×3 convolutions in the E-ecablock module. Using ECA_Block, before each convolution, to obtain meaningful features by assigning attention weights to different channels of the input feature map. Finally, the output result after the convolution and the feature of the initial input is summed through the residual connection, which can obtain a better channel feature. In order to increase the efficiency of channel information extraction, the same method is adopted, adding the ECA_Block module before each layer of convolution in the E-ecaneck module. The channel dimension of the feature map is reduced and added through the ECA_Block module, multiple convolution operations are performed for feature extraction, which can obtain more useful feature information.

3.2. Key Point Estimation and Adaptive Part Generation

Key Point Estimation As shown in Figure 1, there are three fully-connected layers after the convolutional features, with output dimensions 2048, 2048, and *2N*, respectively.

Here N is the number of key points. After each fc layer,the ReLU activation function and drop out layer (ratio 0.5) is used. We use L2 distance loss for key point estimation, $\sum_i^n \|\widehat{p_i} - p_i\|_2^2$, where $\widehat{p_i}, p_i \in R^2$ are the normalized ground truth and estimation for key point i.

Some attributes are clearly associated with certain object parts. We encode such prior knowledge by specifying a subset of key points p_t for each part t. For example, in Figure 8, for body we have P_{body} = {body, head, torso, upbody, lowerbody }. The initial part bounding box $b_t = [w_t, h_t, x_t, y_t]$ is defined as an enlarged bounding box of key points in P_t,

$$w_t = s \cdot (\max_x(p_t) - \min_x(p_t)) \qquad h_t = s \cdot \left(\max_y(p_t) - \min_y(p_t)\right) \quad (4)$$

$$x_t = \frac{1}{2}(\max_x(p_t) + \min_x(p_t) - w_t) \qquad y_t = \frac{1}{2}\left(\max_y(p_t) + \min_y(p_t) - h_t\right) \quad (5)$$

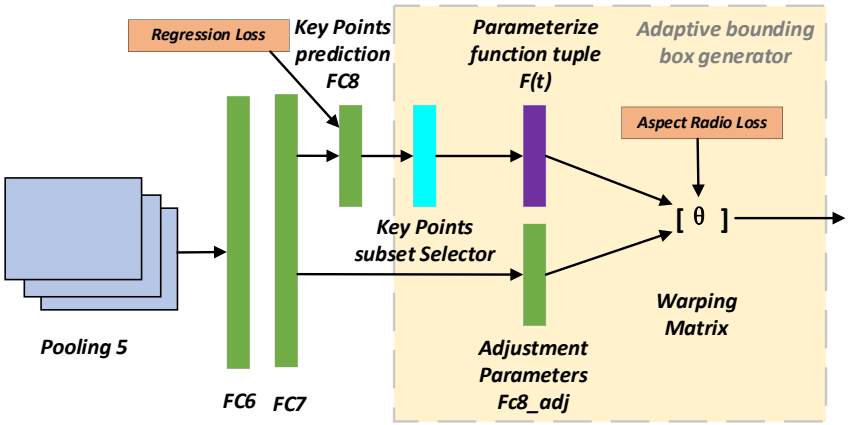


Fig.8. Adaptive bounding box generator

Here, $\max_x(p_t)$is the maximum x coordinate in key points of P_t. Other notations are similar. *w/h* is the initial box's width/height. *x/y* is initial box's upper left corner. *s* is a constant scalar larger than *1.0*. It is set to *1.5* in our experiment. The initial bounding box is then adaptively adjusted by free parameters $\Delta = [\Delta_w, \Delta_h, \Delta_x, \Delta_y]$. The final bounding box is defined as $[w_t(1 + \Delta_w), h_t(1 + \Delta_h), x_t + \Delta_x, y_t + \Delta_y]$. To learn the adjustment parameters, we add one more fully connected layer (fc8_adj) to the previous layer (fc7), with *4* output values. This is depicted in Figure 8. The final bounding box could have too distorted dimensions due to the free adjustment parameters. In order to alleviate this issue, we introduce a bounding box aspect ratio loss as L_r^t

$$L_r^t = \begin{cases} \frac{1}{2}\{[\alpha[h_t(1+\Delta_h)]^2 - w_t(1+\Delta_w)]^2\}_+ & if\ h_t(1+\Delta_h) > w_t(1+\Delta_w) \\ \frac{1}{2}\{[\alpha[w_t(1+\Delta_w)]^2 - h_t(1+\Delta_h)]^2\}_+ & if\ w_t(1+\Delta_w) > h_t(1+\Delta_h) \end{cases} \quad (6)$$

where α is a ratio threshold (set to 0.6 in the experiment). The loss is 0 when the value in bracket $\{\}_+$ is less than 0.

1.1 Hierarchical Pedestrian Attribute Recognition

There are some differences between various pedestrian attributes recognition, that is, some attributes characteristics scattered in various parts of the human body, such as gender, fat or thin, and other attributes characteristics are concentrated in a certain part of the human body, such as hairstyle. According to the characteristics scattered in the whole body or gathered in a local, we divided the pedestrian attributes into global and local. In real deployment, this can be implemented according to the pre-configuration.

Our goal is to recognize a set of human attributes $\{a \in A\}$ for all the people in an image I in unconstrained scenes. Assume that a target person's bounding box in I as b, divide human body b into a set of parts s $\{s \in S\}$ according to human body key points. Probability is used to determine whether an attribute exists, this is, estimated the probability of existence of attribute a on the target person given by measurements V. The attributes are recognized from both human body b and parts S. The measurements V is defined as $V = \{b, S, I\}$. Formula 8 is used to evaluate for each attribute a the conditional probability from the given measurements V:

$$Score(a; b, S) = \omega_{a,b}^{T} \cdot \emptyset(b; I) + \max_{s \in S} \omega_{a,s}^{T} \cdot \emptyset(s; I) \quad (8)$$

where φ(b; I) is the extracted fc7 features from region b in image I, while wa,· are the scoring weights of attribute a for different regions.

The scoring terms for person bounding box b and parts $\{s \in S\}$ form the basis of our model and are shown on the upper two paths in Fig. 8. Their sum can be regarded as a pose-normalized deep representation at the score level. Such score fusion is found to be more effective than feature fusion in our task because the latter would generate a very large feature vector from the many parts and overfits easily. In our CNN, the scoring weights and feature vectors are jointly learned for all attributes $a \in A$.

Note for the part set S, we select the most informative part s for each attribute by a max score operation and only add the maximum to the final attribute score. This is because human attribute signals often reside in different body parts, so not all parts should be responsible for recognizing one particular attribute. For example, the head part can hardly be used to infer the "long pants" attribute. Through the max pooling of part scores, we are now able to capture those distributed attribute signals from the rich part collection.

4. Experiments

In this paper, the model training is divided into two stages: first, key point detection model is trained with COCO2017 dataset, and then attribute recognition model is trained with PETA and RAP dataset.

1.2 Data Set and Evaluation Indicators

COCO2017 dataset has 17 key points of human posture, it contains 200000 images, 250000 images with 17 key points. 57000 images are used for training, 5000 images are used for validate, 20000 images are used for test. The 17 key points are nose, right eye, left eye, right ear, left ear, right shoulder, right elbow, right wrist, left shoulder, left elbow, left wrist, right hip, right knee, right ankle, left hip, left knee, left ankle.

Table 1. Thirty-five attributes of PETA dataset

No	attribute	No	attribute
1	accessory Muffler	19	carryingMessengerBag
2	personalLarger60	20	personalLess45
3	accessoryHat	21	lowerBodyJeans
4	personalMale	22	carryingBackpack
5	hairLong	23	footwearShoes
6	footwearLeatherShoes	24	upperBodyTshirt
7	upperBodyPlaid	25	lowerBodyShortSkirt
8	personalLess60	26	footwearSneaker
9	personalLess30	27	carryingNothing
10	upperBodyShortSleeve	28	upperBodyJacket
11	accessoryNothing	29	carryingOther
12	carryingPlasticBags	30	lowerBodyShorts
13	upperBodyFormal	31	accessorySunglasses
14	upperBodyOther	32	upperBodyThinStripes
15	upperBodyCasual	33	footwearSandals
16	lowerBodyFormal	34	upperBodyLogo
17	lowerBodyCasual	35	upperBodyVNeck
18	lowerBodyTrousers		

Table 2. Fifty-one attributes of RAP dataset

No	attribute	No	attribute
1	Female	27	lb-Dress
2	Clerk	28	attach-Backpack
3	ub-Vest	29	attach-SingleShoulderBag
4	lb-TightTrousers	30	attach-Other
5	lb-Jeans	31	attach-HandBag
6	Customer	32	ub-Jacket
7	shoes-Boots	33	ub-ShortSleeve
8	action-Calling	34	Age31-45
9	lb-LongTrousers	35	Age17-30
10	lb-Skirt	36	action-Pulling
11	hs-LongHair	37	ub-TShirt
12	ub-TShirt	38	shoes-Sport
13	shoes-Leather	39	ub-Tight
14	hs-Glasses	40	attach-PlasticBag
15	action-Pusing	41	action-Gathering
16	hs-Hat	42	attach-PaperBag
17	hs-BaldHead	43	action-Holding
18	hs-BlackHair	44	ub-Sweater
19	AgeLess16	45	hs-Muffler
20	ub-Colton	46	BodyFat
21	ub-SuitUp	47	action-CarrybyArm
22	action-CarrybyHand	48	shoes-Casual
23	attach-HandTrunk	49	BodyThin
24	lb-ShortSkirt	50	BodyNormal
25	shoes-Cloth	51	action-Talking
26	attach-Box		

The PETA dataset consists of 19,000 people images collected from 10 small-scale pedestrian datasets. The entire data set is randomly divided into three non-overlapping parts: 9500 for training, 1900 for verification, and 7600 for testing. Due to the imbalance of attribute samples, attributes with a sample ratio of more than 5% in the 35

attribute labels are generally selected for evaluation. The 35 attributes are shown in Table 1.

The RAP data set contains 41,585 images from 26 indoor monitoring cameras, and each image has 69 binary attributes and 3 multi-category attributes. According to the official agreement, the entire data set is divided into 33,268 training images and 8,317 test images. The recognition performance of 51 binary attributes is evaluated. The 51 attributes are shown in Table 2.

The evaluation indicators for quantitative comparison use the general label-based mean Accuracy (mA) indicator and the Example-based accuracy rate (Accuracy, Acc) indicator and precision rate (Precision, Prec) index, recall rate (Recall, Rec) index and F1 value index.

4.1. Comparative Experiment

This article compares two sets of experiments: Experiment 1 compares the benchmark network Base-CNN and the various modules proposed in this article on the two data set test sets; Experiment 2 compares the model in this article with some current pedestrian attribute recognition models Comparison of quantitative evaluation index results.

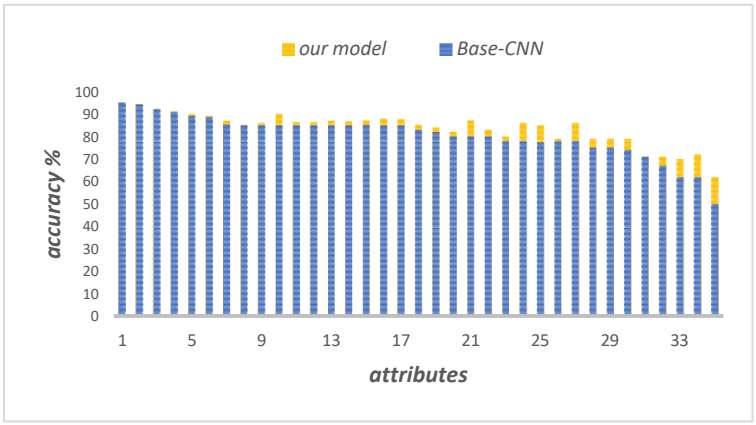
Experiment Related Settings

In this experiment, the COCO2017, RAP, and PETA image data sets will be cropped with the hip of the human body as centre, and adjusted to a fixed ratio of 4:3 for height and width, the size of the image being cropped to 256 × 192. Dataset are used for backbone network training for efficient and accurate human key point detection. In this experiment, the SGD optimizer is selected to optimize the model. The training epochs is set to 200, the training batch size is set to 20, and the learning rate is 0.001. Pedestrian attribute features and key points recognition share the feature map obtained by the backbone network. The feature map of each bounding box is pooled to obtain a 7x7x128 matrix through the corresponding backbone area features. The process of pedestrian attribute recognition training will fix the parameters of the backbone network, and only optimize the parameters of the full link layer. SGD optimization method is used in the training process, with an initial learning rate of 0.001. The whole system is trained and validate based on the pytorch deep learning framework.

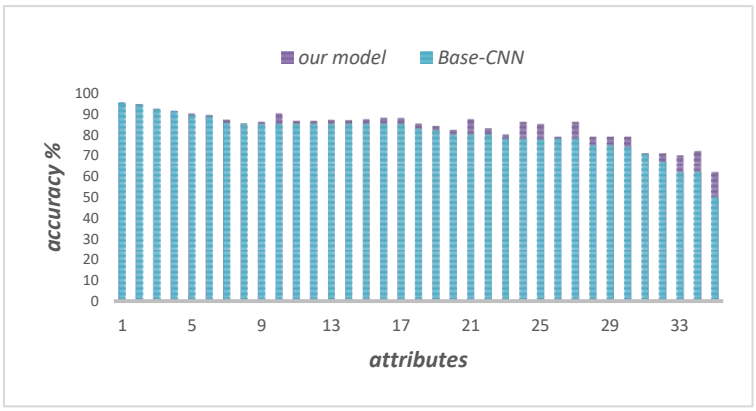
Method validity experiment

In this group of experiments, this paper adds non-local, E-ecablock, and E-ecaneck on the reference network Base-CNN respectively to compare and verify that each method recognizes attributes. The experimental results on the PETA and RAP data sets are shown in Table 3. The content in bold in the table indicates the best result under this index. It can be seen from Table 3 that after the non-local block, on the two data sets, the mA index increased by 1.57 percentage points and 1.94 percentage points respectively compared with the benchmark model; Adding the E-ecablock can also make the recognition effect of the model have a small gain (mA index gain ranges are

0.42 percentage points and 0.26 percentage points respectively); on the basis of adding E-ecaneck module proposed in this article, the value of each evaluation index can be improved in the two data sets, and the mA index has increased by 1.92 and 1.85 respectively. In general, compared with the benchmark model Base-CNN, the mA index on the PETA and RAP data sets has increased by 5.91 percentage points and 6.05 percentage points, respectively. In Figure 9, by comparing the recognition accuracy rates of the benchmark model Base-CNN and this model in the 35 attributes of the PETA dataset and the 51 attributes of the RAP dataset, it can be seen that the model in this paper has a recognition effect on most attributes. The improvement, especially attributes with a small number of samples in dataset, is more obvious. The Y-axis in Figure 4 represents the accuracy of recognition, and the X-axis represents the attribute number, which corresponds to the attribute numbers in Table 1 and Table 2, respectively.



(a) PETA



(b) RAP

Fig. 9. The recognition accuracy of each attribute on PETA and RAP of the benchmark model and our model

Table 3. Effectiveness comparison of different modules on PETA dataset

Model	mA	Prec	Rec	F1
Base-CNN	80.72	85.42	81.76	83.55
Base-CNN +non_local	82.29	83.41	82.49	82.95
Base-CNN+non_local+ E-ecablock	82.71	83.87	84.21	84.04
Base-CNN +non_local + E-ecablock + E-ecaneck	**86.63**	**86.79**	**85.73**	**86.26**

Table 4. Effectiveness comparison of different modules on RAP dataset

Model	mA	Prec	Rec	F1
Base-CNN	76.65	76.93	76.93	76.93
Base-CNN +non_local	78.59	78.16	78.85	78.5
Base-CNN+non_local+ E-ecablock	78.85	78.36	79.25	78.8
Base-CNN +non_local + E-ecablock + E-ecaneck	**82.70**	**81.85**	**83.82**	**82.82**

Comparison with other models

Talbe 5. Effect comparison of different models on PETA unit:%

model	mA	Prec	Rec	F1
DeepMAR	82.89	83.68	83.14	83.41
HPNet	81.77	84.82	83.24	84.02
PGDM	82.97	86.86	84.68	85.76
MPAR	83.57	86.19	85.07	85.63
IA^{2}Net	84.13	85.73	**86.07**	85.90
Our model	**86.63**	**87.79**	85.73	**86.75**

Talbe 6. Effect comparison of different models on RAP unit:%

model	mA	Prec	Rec	F1
DeepMAR	73.79	74.92	76.21	75.56
HPNet	76.12	77.33	78.79	78.05
PGDM	74.31	78.86	75.90	77.35
LGNet	78.68	80.36	79.82	80.09
MPAR	75.64	78.29	79.23	78.76
VSGR	77.91	82.05	80.64	81.34
RCRA	78.47	**82.67**	76.65	79.55
IA^{2}Net	77.44	79.01	77.45	78.22
Our model	**82.70**	81.85	**83.82**	**82.82**

Many models are based on PETA and RAP dataset for pedestrian attribute recognition, mainly including PGDM (Posed Guided Deep Model) [13], VSGR (Visual-Semantic Graph Reasoning net) [36], HPNet (Hydra Plus Net) [25], DeepMAR (Deep Multi-Attribute Recognization model) [12], LGNet (Location Guided Network) [37], MPAR (Multistage Pedestrian Attribute Recognition method) [38], RCRA (Recurrent Convolutional and Recurrent Attention model) [39] And IA^{2}Net (Image-Attribute reciprocally guided Attention Network) [40]. Among them, DeepMAR only extracts the global features of pedestrians. PGDM and LGNet extract the local features of different parts of pedestrians through the pedestrian local area network. RCRA and IA^{2}Net apply the attention mechanism to the network model. The bold font in the table 5 and 6 means

the best result. From Table 5 and Table 6, we can see that our model has strong competitiveness in mean accuracy, precision and F1 in PETA and RAP dataset. The mean accuracy of attribute recognition on PETA and RAP datasets reached 86.63% and 82.70%, superior to the existing models.

Sample recognition analysis:

Figure 10 shows 8 images labelled *a*-*h*. The histogram is the prediction results of our model and StrongBaseline [41]. The horizontal axis of the histogram represents the predicted attribute, the vertical axis represents the probability of predicted attribute. The blue bar is the probability predicted by the our model, and the orange bar is the probability predicted by the StrongBaseline model. For the attributes of clear samples, such as samples *a*, *c*, and *e*, both methods have excellent and similar prediction results. For the attributes of blurry and occluded samples, such as the *g* sample is blurry and the 'ShoulderBag' attribute is occluded by body, the prediction probabilities of our model and StrongBaseline are only 0.0033 and 0.0027, which are shown zero on the histogram. In addition, Not only we found that the attribute with high resolution, such as 'Back', 'UpperLogo', 'Trousers', 'Age', ' Side', etc. have excellent prediction probability by our model, but also the 'Back' attribute of sample *d* and *f*, the 'Side' attribute of sample *g* and *h*, the prediction results of our model are much higher than StrongBaseline. Based on these, we infer that the spatial self-attention module in our model network plays an important role in capturing the global contextual information and improving the recognition accuracy of the model.

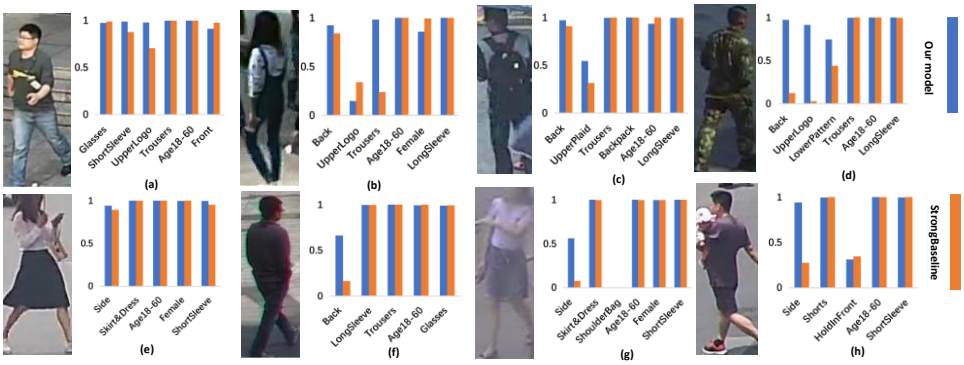


Fig.10. Comparison of pedestrian attribute recognition accuracy. The accuracy of our model is significantly higher than StrongBaseline on low quality image

Sample heatmap analysis:

To better understanding where the model focuses on, we visualize these localization results in Figure 11, which shows randomly selected samples on the RAP dataset. Based on our model: (1) The two groups *a* and *b* are 'Skirt' positive and 'BackPack' positive, respectively. Samples *a.1* and *b.1* are the areas that the main network Base-CNN focus

on, *a.2* and *b.2* are the areas that are concerned after dual self-attention. We can observe that the attribute regions of interest are more complete after dual self-attention model. (2) The two groups *c* and *d* are 'LongSleeve' negative and 'BackPack' negative, respectively. *c.1* and *d.1* are the areas of interest for the main network Base-CNN, *c.2* and *d.2* are the regions of interest after dual self-attention model. We observe that the former focuses on invalid regions, while the latter does not pay attention to any regions. From Figure 12 the results can be seen, (1) Group *e* is 'Skirt' positive; *e.1* and *e.2* are the regions concerned by StrongBaseline and our model. We can observe that most of the regions concerned by StrongBaseline are invalid, while our model basically pays attention to all regions of the 'Skirt'.(2) Group *f* is 'BackPack' negative. Similarly, *f.1* and *f.2* are the regions that StrongBaseline and our model focus on respectively. As analysed above, StrongBaseline pays attention to an invalid region, while our model does not focus on any regions. According to these, we infer that our model will first predict an attention region for positive or negative attribute, and then strengthen the attention region for positive attribute or weaken the attention region for negative attribute by dual self-attention, such as groups *a* and *c*.

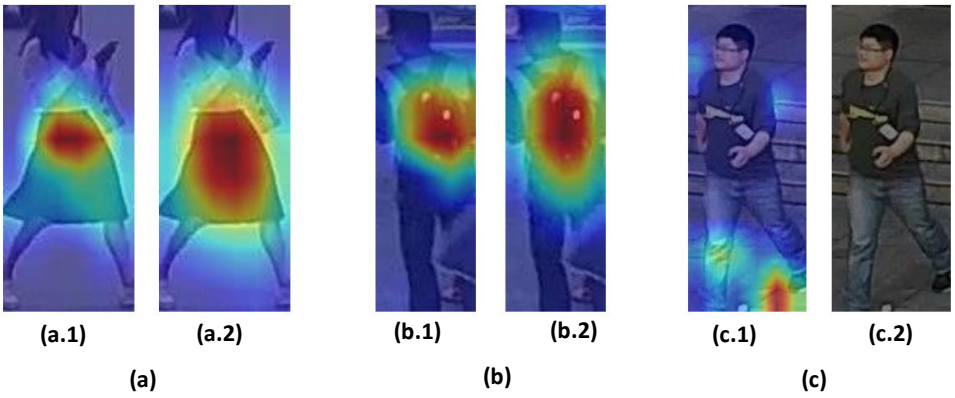


Fig.11. Heatmap of Base-CNN and our model on samples.

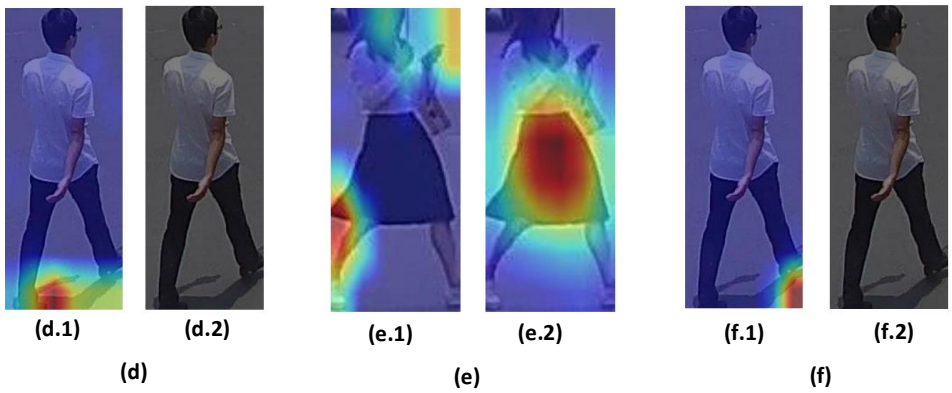


Fig.12. Heatmap of StrongBaseline and our model on samples.

5. Conclusions

According to the semantic correlation between pedestrian attributes and the spatial correlation between each attribute and body parts, this paper proposes to use spatial self-attention mechanism to extract spatial dependent features, and the channel self-attention mechanism to extract semantic relevance of features between channels. Use Non-Local network structure to highlight important features and weaken unimportant features Two attention modules, E-eaclock and E-ecanenck, are designed as the basic modules to form a feature extraction network, which can effectively improve the representation ability of spatial and semantic features. The key points of pedestrians and five bounding boxes of body part are extracted through feature extraction in the previous stage. The accuracy of attribute recognition is improved by making full use of attribute space and semantic constraints in attribute recognition. The experimental results show that the model in this paper can improve the recognition effect of attributes in both the PETA and RAP pedestrian attribute data sets. Through the analysis of the heat map of the experimental results, it can be seen that our model will first predict an attention region for positive or negative attribute, and then strengthen the attention region for positive attribute or weaken the attention region for negative attribute by dual self-attention. This method is important for pedestrian attribute recognition and can be applied to various practical applications.

Acknowledgment. This research work was supported in part by National Key R&D Program of China (2019YFC1520500, 2020YFC1523004).

References

1. Gordo A, Almazan J, Revaud J, et al. End-to-end learning of deep visual representations for image retrieval[J]. International Journal of Computer Vision, 2017, 124(2): 237-254.
2. Dubey S R. A decade survey of content based image retrieval using deep learning[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2021.
3. Cffa B , Gg A , Jvdw A , et al. Saliency for fine-grained object recognition in domains with scarce training data[J]. Pattern Recognition, 2019, 94:62-73.
4. Liang, M., & Hu, X. (2015). Recurrent convolutional neural network for object recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3367-3375).
5. Ding, S., Lin, L., Wang, G., & Chao, H. (2015). Deep feature learning with relative distance comparison for person re-identification. Pattern Recognition, 48(10), 2993-3003.
6. Ryan Layne, Timothy M Hospedales, Shaogang Gong, and Q Mary. Person re-identification by attributes. In Bmvc, volume 2, page 8, 2012.
7. Jianqing Zhu, Shengcai Liao, Dong Yi, Zhen Lei,and Stan Z Li. Multi-label cnn based pedestrian attribute learning for soft biometrics. In ICB. IEEE, 2015.
8. Yubin Deng, Ping Luo, Chen Change Loy, and Xiaoou Tang. Pedestrian attribute recognition at far distance. In ACM MM, 2014..
9. Y. Deng, P. Luo, C. C. Loy, and X. Tang. Pedestrian attribute recognition at far distance. In Proc. ACM Multimedia, 2014.
10. R. Layne, T. M. Hospedales, S. Gong, and Q. Mary. Person reidentification by attributes. In Proc. BMVC, 2012.

11. J. Zhu, S. Liao, Z. Lei, D. Yi, and S. Z. Li. Pedestrian attribute classification in surveillance: Database and evaluation. In Proc. ICCV Workshops, 2013.
12. D. Li, X. Chen, and K. Huang, "Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios," in Pattern Recognition (ACPR), 2015 3rd IAPR Asian Conference on. IEEE, 2015, pp. 111–115.
13. D. Li, X. Chen, Z. Zhang, and K. Huang, "Pose guided deep model for pedestrian attribute recognition in surveillance scenarios," in 2018 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2018, pp. 1–6.
14. N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In Proc. CVPR,2014.
15. L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations," in Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009, pp. 1365–1372.
16. D. A. Vaquero, R. S. Feris, D. Tran, L. Brown, A. Hampapur, and M. Turk. Attribute-based people search in surveillance environments. In Workshop on Applications of Computer Vision (WACV), pages 1–8, 2009.
17. J.Q. Zhu, S.C. Liao, Z. Lei, D. Yi, S.Z. Li. Pedestrian Attribute Classification in Surveillance: Database and Evaluation. IEEE ICCV Workshop, 2013.
18. J.Q. Zhu, S.c. Liao, D.Yi, Z. Lei, S.Z. Li. Multi-Label CNN Based Pedestrian Attribute Learning for Soft Biometrics. IAPR ICB, 2015.
19. Y.Deng, P.Luo, c.c. Loy, X.O. Tang. Pedestrian Attribute Recognition at Far Distance. ACM MM, 2014.
20. W. Chen, X. Chen, J. Zhang, and K. Huang. A multi-task deep network for person re-identification. In AAAI, 2017.
21. P. Sudowe, H. Spitzer, B. Leibe. Person Attribute Recognition with a Jointly-trained Holistic CNN Model. IEEE ICCV Workshop, 2015.
22. A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, "Multi-task cnn model for attribute prediction," IEEE Transactions on Multimedia, vol. 17, no. 11, pp. 1949–1959, 2015.
23. Yao, C. , et al. "Hierarchical pedestrian attribute recognition based on adaptive region localization." IEEE International Conference on Multimedia & Expo Workshops IEEE, 2017.
24. Feng, Z. , et al. "Learning Spatial Regularization with Image-Level Supervisions for Multi-label Image Classification." IEEE (2017).
25. X. Liu, H. Zhao, M. Tian, L. Sheng, J. Shao, S. Yi, J. Yan, and X. Wang,"Hydraplus-net: Attentive deep features for pedestrian analysis," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 350–359.
26. Li, Qiaozhe, et al. "Pedestrian Attribute Recognition by Joint Visual-semantic Reasoning and Knowledge Distillation." IJCAI. 2019.
27. Huang, Lingli. "Improved non-local means algorithm for image denoising." Journal of Computer and Communications 3.04 (2015): 23.
28. Wang, X., Girshick, R., Gupta, A., & He, K. (2018). Non-local neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7794-7803).
29. Zhang, Han, et al. "Self-attention generative adversarial networks." International conference on machine learning. PMLR, 2019.
30. Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems. 2017.
31. Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
32. Woo, Sanghyun, et al. "Cbam: Convolutional block attention module." Proceedings of the European conference on computer vision (ECCV). 2018.
33. Chen, Tianlong, et al. "Abd-net: Attentive but diverse person re-identification." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

34. Tan, Zichang, et al. "Attention-based pedestrian attribute analysis." IEEE transactions on image processing 28.12 (2019): 6126-6140.
35. Yang, Z. , et al. "Gated Channel Transformation for Visual Recognition." 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, 2020.
36. Q. L. X. Z. R. H. K. HUANG, "Visual-semantic graph reasoning for pedestrian attribute recognition," in Association for the Advancement of Artificial Intelligence, AAAI, 2019.
37. P. Liu, X. Liu, J. Yan, and J. Shao, "Localization guided learning for pedestrian attribute recognition" 2018.
38. Zheng S F, Tang J, Luo B, et. al. Multistage pedestrian attribute recognition method based on improved loss function. Pattern Recognition and Artificial Intelligence.2018.31(12):1085-1095.
39. Zhao X, Sang L, Ding G, et al. Recurrent attention model for pedestrian attribute recognition. AAAI Press, 2019.
40. Ji Z, He E, Wang H et al. Image-attribute reciprocally guided attention network for pedestrian attribute recognition. Pattern Recognition Letters.2019, pp. 89-95.
41. J. Jia, H. Huang, W. Yang, X. Chen, and K. Huang,"Rethinking of Pedestrian Attribute Recognition: Realistic Datasets with Efficient Method," arXiv preprint arXiv:2005.11909, 2020.

Zhongkui Fan receive the BS degree in Computer engineering from Huanghe science and technology College, and the MS degree in Computer Engineering from Jiangxi University of Science and Technology. He is currently working toward the PhD degree in the School of Communication and Information Engineering with Shanghai University. His research interests include computer vision and machine learning.

Ye-peng Guan was born in Xiaogan, Hubei Province, China, in 1967. He received the B.S. and M.S. degrees in physical geography from the Central South University, Changsha, China, in 1990, 2006, respectively, and the Ph.D. degree in geodetection and information technology from the Central South University, Changsha, China, in 2000. From 2001 to 2002, he did his first postdoctoral research at Southeast University in electronic science and technology. From 2003 to 2004, he did his second postdoctoral research at Zhejiang University in communication engineering, and he had been an Assistant Professor with the Department of Information and Electronics Engineering, Zhejiang University. Since 2007, he has been a Professor with School of Communication and Information Engineering, Shanghai University. He is the author of more than 120 articles, and more than 20 patents. His research interests include intelligent information perception, digital image processing, computer vision, and security surveillance and guard.

Received: August 15, 2022; Accepted: January 03, 2023.

Computer Science and Information Systems 20(2):813–829 https://doi.org/10.2298/CSIS220808022Y

Inverse Halftoning Based on Sparse Representation with Boosted Dictionary

Jun Yang[1], Zihao Liu[1], Li Chen[1], Ying Wu[2], and Gang Ke[3]

[1] College of Information Science and Engineering, Jiaxing University, Jiaxing, China
{juneryoung,lzh}@zjxu.edu.cn, chenli20040415@163.com
[2] College of Information Engineering, Jiaxing Nanhu University, Jiaxing, China
wy43601@163.com
[3] Department of Computer Engineering Dongguan Polytechnic, Dongguan, China
403513673@qq.com

Abstract. Halftone image is widely used in printing and scanning equipment. It is significant for the halftone image to be preserved and processed. For the different resolution of the display devices, the processing and displaying of halftone image are faced with great challenges, such as Moore pattern and image blurring. The inverse halftone technique is required to remove the halftone screen. In this paper, we propose an inverse halftone algorithm based on sparse representation with the dictionary learned by two steps: deconvolution and sparse optimization in the transform domain to remove the noise. The main contributions of this paper include three aspects: first, we analysis the denoising effects for different training sets and the dictionary; Then we propose the denoising algorithm through adaptively learning the dictionary, which iteratively remove the noise of the training set and improve the dictionary; Then the inverse halftone algorithm is proposed. Finally, we verify that the noise level in the error diffusion linear model is fixed, and the noise level is only related to the diffusion operator. Experimental results show that the proposed algorithm has better PSNR and visual performance than state-of-the-art methods. The codes and constructed models are available at https://github.com/juneryoung2022/IH-WNNM.

Keywords: inverse halftoning, deconvolution, sparse representation, error diffusion.

1. Introduction

Halftone technology converts a continuous tone image into the binary version. Image halftoning can be seen as a map from R^N to $\{0,1\}^N$, where N is the dimension of image sizes. Image halftone algorithm is widely used in the scanning and printing process of newspaper, book, magazine and fax. Ordered dithering and error diffusion is the two main methods of halftoning [7,11], Fig. 1 (a) shows the halftone images that generated from the continuous tone color image with size of $256 \times 256 \times 3$ by using ordered dithering filter, which is the left part of Eq. (1). While Fig. 1 (b) and (c) are the halftone version by using error diffusion methods with Floyd and Jarvis halftone filters respectively. In the ordered dithering method, the halftone image is generated by comparing the pixels between the original image and the filter matrix correspondingly. The error diffusion technique compares the image pixel value with a fixed threshold value, and diffuses the quantization

error into the neighbor pixels according to the weights. Floyd and Jarvis proposed the error diffusion algorithms with different filter generators [6, 9]. which are the middle and right parts of Eq. (1) respectively. Kite et al. proposed the linear model that simulate the error diffusion filter, and pointed out that error diffusion halftone image can be regarded as a convoluted continues tone image plus the noise. It is necessary to inverse the halftone image to continuous tone for the halftone version does not contain detail information of image and with bad visual defect. And the halftone image is not easy to be processed. For example, the halftone image should be inverse halftoned in the scanning process. In this paper, we proposed an effective inverse halftone algorithm, which reconstructs the continuous tone image form the halftone version.

(a) (b) (c)

Fig. 1. The ordered dithering halftone image (a), Floyd and Jarvis error diffusion halftone images (b) and (c) generated from the color *Lena* image with the filters in Eq. (1) respectively

$$(1/17)\times\begin{bmatrix}8 & 11 & 7 & 10\\ 15 & 1 & 64 & 4\\ 6 & 9 & 5 & 12\\ 13 & 3 & 14 & 2\end{bmatrix},\quad (1/16)\times\begin{bmatrix}0 & \bullet & 7\\ 3 & 5 & 1\end{bmatrix},\quad (1/48)\times\begin{bmatrix}0 & 0 & \bullet & 7 & 5\\ 3 & 5 & 7 & 5 & 3\\ 1 & 3 & 5 & 3 & 1\end{bmatrix}\qquad (1)$$

2. Related work

There are not effective inverse halftone methods and the noise removal algorithms are still the normally traditional approaches. Gaussian low-pass filtering inverse halftone [3] is a simple and fast method, but it cannot retain image edge and texture information effectively. The wavelet based methods reconstruct the inverse halftone image by transforming and processing the image in the wavelet domain [13, 22], in which the literature [22] divides the inverse halftone procedure into two steps: deconvolution and denoising. Literature [16] proposed the nonlocal regularization inverse halftone method, which contains two regularization procedure: the first one is total variation (TV) regularization based on the BM3D denoising algorithm; the second one is the post-processing by using the

non-local regularization. The literature [25] descreens the image through training the descreening model, which contains two nonlinear operators: one is resolution synthesis-based denoising (RSD); and the other is SUSAN filtering. Literature [26] uses trained double dictionaries to reconstruct the continuous tone image. The two dictionaries are trained by the continuous image and the halftone version respectively, which using the same coefficient. The coefficient getting by the halftone image multiplies the dictionary trained using the continuous image, which can directly output the inverse halftone image. With the successful application of deep learning in image processing, the neural-network based methods have also been proposed and used in image inverse halftone reconstruction [8,10,15,17,24,27,28], which are in full using the prior information of inner images. Literature [34] address the inverse Halftone Colorization task, which tries to recover colorful images from black and white halftone prints, and can be treated as the joint problem of inverse halftone and colorization. In this literature the inverse halftone network architecture is extended from the state-of-the-art PRL model [32] where its content aggregation for synthesizing the global tone is replaced by a 7-layers U-Net to better support the analog halftoning. These methods directly train a map from halftone image to continuous tone image. And the map is used to inverse the halftone image. Sparse based methods [19,35], feature selection or fusing methods [36, 37], Seeded random walk [30] and evolutionary algorithm [18] are machine learning based approaches, which show their well performance in classification and optimization problems. The approaches of machine learning and deep learning can directly inverse halftoning images without known the halftone filters. Although the results of these methods are well, but the training procedure is usually time consuming.

For the above problems, we need to find an effective algorithm to remove the halftone noise. Kite proposed the linear model of error diffusion and points out that the error diffusion halftone images can be seen as the combination of continuous tone image and add noises [11, 12]. Then the continuous tone image is obtained after denoising. There are three contributions in this paper listed as follows.

Firstly, we proposed an adaptive denoising algorithm based on sparse representation with trained dictionaries. By removing the noise of training image patches, the cleaning dictionaries can be output. In this way, both the cleaner dictionary and image can be got for several times training. Secondly, we proposed an inverse halftone algorithm. Similarly with the literature [22], we deconvolute the halftone image and output the continuous tone image with noise. Then we denoise the image by using the updated dictionaries in an iterative way. Thirdly, the gain in the linear error diffusion model proposed by Kite is a constant, which only relative with the operator of the linear model. We point out and verify experimentally that the noise level (denoted as σ) of the halftone image is also a constant.

The reasons for the better performance of our method can be explained that the clean image patches can get clean dictionary with good sparse representation ability compared with wavelet transformations [1,5]. And the output continuous image is the combination of the dictionary atoms. It is obviously that cleaner dictionary can be result in clean image. Here we use the patches of the denoised image to train the dictionary adaptively, where the patches size is 8×8 (the dimension size of the atom is 64). We improve the denoising performance for the image by removing the noise in the dictionary in each iteration.

3. Error diffusion linear model and deconvolution

3.1. Error diffusion model

Error diffusion halftone method is to quantify the gray image into halftone image with only two gray levels. The error diffusion model proposed by Floyd is shown as Fig. 2 (a), where $x(n_1, n_2)$ is the continuous tone image, (n_1, n_2) denotes the coordinates, $z(n_1, n_2)$ is the halftone image, $x'(n_1, n_2)$ is generated by diffusing the error $e(n_1, n_2)$ to the neighbor of $x(n_1, n_2)$ by the filter H. And H is the impulse response of $h(n_1, n_2)$. Different H can result in different error diffusion algorithms, such as Floyd and Jarvis error diffusion algorithm. The quantization error is diffused according to the weight in the neighbor field of the dot. The quantization model is nonlinear for using the quantizer to generate the halftone image. Kite et al. proposed the approximate linear model, which using a gain module K and additive white noise $\gamma(n_1, n_2)$ to simulate the effects of quantizer. The linear model is shown as Fig. 2 (b), using which the halftone image $z(n_1, n_2)$ can be denoted by the gray level image $x(n_1, n_2)$ and the additive white noise $\gamma(n_1, n_2)$.

$$z(n_1, n_2) = Px(n_1, n_2) + Q\gamma(n_1, n_2) = (p * x)(n_1, n_2) + (q * \gamma)(n_1, n_2) \quad (2)$$

where $*$ denotes the convolution operator, P and Q denote the linear time invariant (LTI) system.

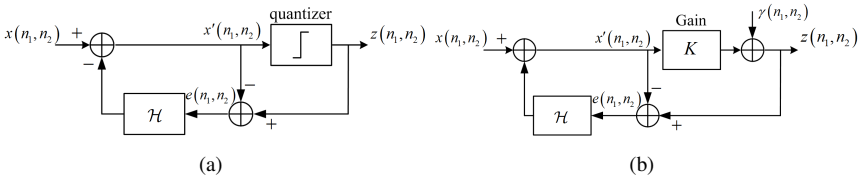


Fig. 2. Error diffusion model. The former is nonlinear error diffusion and the other is linear one

3.2. Halftone image deconvolution

The inverse halftone can be seen as a classic deconvolution problem. With the noise level $Q\gamma(n_1, n_2)$, The gray image $x(n_1, n_2)$ can be obtained from the deconvolution of the halftoning image $z(n_1, n_2)$ and the filter operator P. Kite et al. proposed the linear model of error diffusion.

$$Z(f_1, f_2) = \frac{KX(f_1, f_2) + (1 - H(f_1, f_2))R(f_1, f_2)}{1 + (K - 1)H(f_1, f_2)} \quad (3)$$

when

$$P(f_1, f_2) = \frac{K}{1 + (K - 1)H(f_1, f_2)}, \quad Q(f_1, f_2) = \frac{1 - H(f_1, f_2)}{1 + (K - 1)H(f_1, f_2)}$$

then

$$Z(f_1, f_2) = P(f_1, f_2)X(f_1, f_2) + Q(f_1, f_2)R(f_1, f_2) \tag{4}$$

from Eq. (4) we can get the following equation.

$$P^{-1}(f_1, f_2)Z(f_1, f_2) = X(f_1, f_2) + P^{-1}(f_1, f_2)Q(f_1, f_2)R(f_1, f_2) \tag{5}$$

Transforming Eq. (5) in Fourier domain, we can get the Eq. (6)

$$P^{-1}z(n_1, n_2) = x(n_1, n_2) + P^{-1}Q\gamma(n_1, n_2) \tag{6}$$

where the $P(f_1, f_2)$ and $Q(f_1, f_2)$ are the responses of P and Q in the frequency domain, which are the transformed functions of the signal and the noising respectively. $P(f_1, f_2)$, $Q(f_1, f_2)$, $H(f_1, f_2)$, $X(f_1, f_2)$, $Z(f_1, f_2)$ and $R(f_1, f_2)$ are the 2D Fourier transformation of $p(n_1, n_2)$, $q(n_1, n_2)$, $h(n_1, n_2)$, $x(n_1, n_2)$, $z(n_1, n_2)$ and $\gamma(n_1, n_2)$ respectively. Giving the error diffusion methods, such as the $h(n_1, n_2)$, K is constant for different images [12]. Such as $K \approx 2.03$ for the Floyd error diffusion method and $K \approx 4.45$ for Jarvis method. We will verify that the noise $\gamma(n_1, n_2)$ and the noise level are both constants for the error diffusion method. Inverse halftone can be regarded as the deconvolution of halftone image. The deconvolution algorithm of halftone image includes the following two steps.

Step 1: Convolution operation:

According to Eq. (2) and Eq. (3), the estimate noise image $y(n_1, n_2)$ of the input $x(n_1, n_2)$ can be got.

$$y(n_1, n_2) = P^{-1}z(n_1, n_2) = x(n_1, n_2) + P^{-1}Q\gamma(n_1, n_2) \tag{7}$$

Step 2: Denoising in the transform domain:

The Energy of the discontinuous edge of the image is distributed in many Fourier transform coefficients, while the deconvolution technology based on Fourier transform will result in ringing and blurring artifacts. This is the uneconomical representation of Fourier transform, where the coefficients are easy to confusion with noise. And the wavelet transform can provide a compact image edge sharpening representation. Therefore Neelamani et al. used the Wavelet transform to denoise [22]. The sparse representation based on the compact dictionary, which is more economical comparing the wavelet one [1, 5]. We inverse the halftone image by using the sparse representation method based on the dictionary learning by using the K-SVD algorithm. The inverse halftone model can be shown in Fig. 3

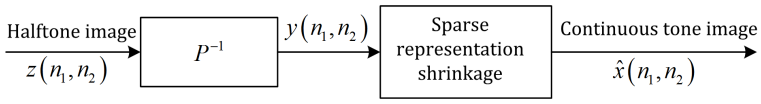


Fig. 3. Image inverse halftoning model

4. Image denoising based on sparse representation

4.1. Sparse and redundant representation of images

In recent years, the theory and application of sparse representation have been developed continuously [4, 23]. Sparse representation has achieved good results in such aspects as image restoration [21], denoising [5], super-resolution [33], face recognition [31] and so on. After transforming under the basis function, the signal can be expressed as a few non-zero coefficients, while most of the coefficients are zero or close to zero, then the signal is sparse or sparsely represented. The basic functions include Fourier transform (FFT), discrete cosine transform (DCT) and various wavelet transform. The method of optimal directions (MOD) has drawn renewed attention [2], which training a dictionary by using the signal vectors. The K-SVD algorithm [1] can train an completely dictionary adaptively and has a good sparse signal representation capability. Some methods such as sparse coding proposed by H. Lee [14], online dictionary learning proposed by Julien [20] and so on, can improve the calculation speed while maintaining well sparse capability.

4.2. Image patch denoising based on sparse representation

The literature [5] uses the image patches with size $\sqrt{n} \times \sqrt{n}$, $n = 8$, to construct a redundant dictionary with size $n \times k$, $k > n$. The sparse representation model show that each image patch x can be sparse represented

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \quad s.t. \quad D\alpha \approx x \tag{8}$$

where the coefficient $\hat{\alpha}$ is sparse, that is $\|\alpha\|_0 \ll n$. The $\|\alpha\|_0$ denotes the number of non-zeros of α.

Assuming that x is sparse and polluted by additive zero-mean Gaussian noise of mean square error σ, the corresponding noisy image block y is obtained. For this image block, the following equation can be used to denoise

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \quad s.t. \quad \|D\alpha - x\|_2^2 \leq T \tag{9}$$

where T is defined by ε and σ, and the denoised image is $\hat{x} = D\alpha$. Eq. (9) becomes the following form by changing the restriction to the penalty term

$$\hat{\alpha} = \arg\min_{\alpha} \|D\alpha - y\|_2^2 + \mu\|\alpha\|_0 \tag{10}$$

The above optimization problem is an NP hard problem. Matching and Baise Pursuit algorithms [2] can be very efficient in obtaining an approximate solution. Since Orthonormal Matching Pursuit (OMP) algorithm [29] is simple and effective, this paper chooses OMP algorithm to solve it.

4.3. Image denoising based on sparse representation

Based on the model in above section the image X with size of $\sqrt{N} \times \sqrt{N}$,$N \gg n$, can be denoised by dividing into $(\sqrt{N} - \sqrt{n} + 1) \times (\sqrt{N} - \sqrt{n} + 1)$ patches. The image

Algorithm 1 Image Denoising Based on Sparse Representation

Input: Giving image Y with addition Gauss white noise (the standard variance is σ).

Parameters: m is the number of iterations, and λ is the Lagrange operator.

1: Initialization: $X = Y$, $D =$ complete DCT dictionary.

2: Updating the following two phases alternately for m times.

- *update the sparsity coefficient* $\{\alpha_{ij}\}$: suppose D and X are constant, resolve the following formulation by using OMP algorithm, get the sparse coefficients of each image patches.

$$\hat{\alpha}_{ij} = \arg\min_{\alpha} \|D\alpha - R_{ij}X\|_2^2 + \mu\|\alpha\|_0 \tag{12}$$

- *update the dictionary* D: suppose X is constant, updating both the atom of the dictionary D and the corresponding coefficient $\{\alpha_{ij}\}$ by decomposing the singular values of the error matrix.

3: updating the image X: iteratively updating $\{\alpha_{ij}\}$ and D, the denoised image can be output by optimizing the formulation

$$\hat{X} = \arg\min_{X} \|X - Y\|_2^2 + \sum\nolimits_{ij} \|D\alpha_{ij} - R_{ij}X\|_2^2 \tag{13}$$

Eq. (13) is a simple quadratic term

$$\hat{X} = (\lambda I + \sum\nolimits_{ij} R_{ij}^T R_{ij})^{-1}(\lambda Y + \sum\nolimits_{ij} R_{ij}^T D\hat{\alpha}_{ij}) \tag{14}$$

Output: $\hat{X}$

denoising problem can be solved through optimizing the following equation with three penalty terms [5].

$$\{\hat{\alpha}_{ij}, \hat{X}\} = \arg\min_{\alpha_{ij},x} \lambda\|X - Y\|_2^2 + \sum\nolimits_{ij} \mu_{ij}\|\alpha_{ij}\|_0 + \sum\nolimits_{ij} \|D\alpha_{ij} - R_{ij}X\|_2^2 \tag{11}$$

where the first term requests that the observing image Y is approximate equal with the denoised image X, and the second term requests that the number of non-zero coefficients, and μ_{ij} is the pre-defined weight of each image patch, and the last term request that each reconstructed patch $R_{ij}X$ can be denoted by the dictionary D and the coefficients α_{ij}, where the matrix R_{ij} denotes the picked patch in the (i, j) of the image. The denoised image can be output by optimizing Eq. (11). The dictionary D is important for the efficiency of the algorithm. The literature [5] proposed a denoising algorithm based on the sparse representation of image patches, which described as follow.

5. Image denoising and inverse halftoning

5.1. Image denoising based on adaptive sparse representation

We consider to improve the performance of the denoising algorithm by improving the dictionary quality. Based on the algorithm 1, an improved sparse presentation denoising algorithm is proposed. Adaptive dictionary learning refers to the use of noisy images patches as a training set to learn the dictionary [5]. The learned dictionary contains noise because the atoms of the initial dictionary are directly coming from the image training

Fig. 4. Nature image training set

set. We remove the noise in the training samples in a feedback way. The quality of the dictionary is gradually approaching to the ground truth training dictionary (GTD).

The sizes (or the redundancy) of the dictionary and the training set directly impact on the image denoising effect. For the image training set without noising, with the increasing of dictionary redundancy the denoising performance tends to be enhanced. It can be seen in Fig. 5 the two curves of denoising performance for GTD and global dictionary (GD) methods. For the image training set with noise, the bigger size of dictionary the more noise in the dictionary. When the redundancy of dictionary increases to a certain balance point, the performance of denoising tends to decrease. It can be seen in Fig. 5 the curves of denoising performance for adaptive dictionary (AD) method. In the improved denoising algorithm, the noise in the training set is removed iteratively in order to clean training set, which can approximate the ground truth (GT) image training set. At the same time, we increase the dictionary redundancy to enhance the denoising ability.

Using the noise image Y to training the dictionary, Abaron et al. proposed the K-SVD algorithm [1] to sparsely encode the training samples $\{y_i\}_{i=1}^{N}$ and update the atoms of the dictionary. The trained dictionary D can be generated by the following formula

$$\min_{D,\alpha}\{\|Y - D\alpha\|_F^2\} \quad s.t. \quad \forall i, \|\alpha_i\|_0 \leq T_0 \tag{15}$$

The training process is divided into two stages: sparse coding stage and dictionary atomic update stage. The proposed algorithm employs the noise image as the training set to train the dictionary, and then using the dictionary to denoise the image. The denoised image as the new version to training the new dictionary and so on.

$$\min_{\alpha}\{\|X - D^{(J)}\alpha\|_F^2\} \quad s.t. \quad \forall i, \|\alpha_i\|_0 \leq T_0 \tag{16}$$

where $\hat{X}^{(J)}$ and $D^{(J)}$ are the image and dictionary in the Jth training and denoising. The chart of the algorithm is shown in Fig. 6. Experiments results show that the quality of the dictionary is boosted and gradually stable in the first several iterations. The performance of image denoising is also boosted with the updated dictionary. The algorithm is shown as Algorithm 2.

5.2. Image inverse halftoning based on adaptive dictionary training

The proposed algorithm 2 is divided into two stages. The diagram is shown in Fig. 3, where Y and $\hat{X}^{(J)}$ are the same as $y(n_1, n_2)$ and $\hat{x}(n_1, n_2)$ respectively. Here Y is the

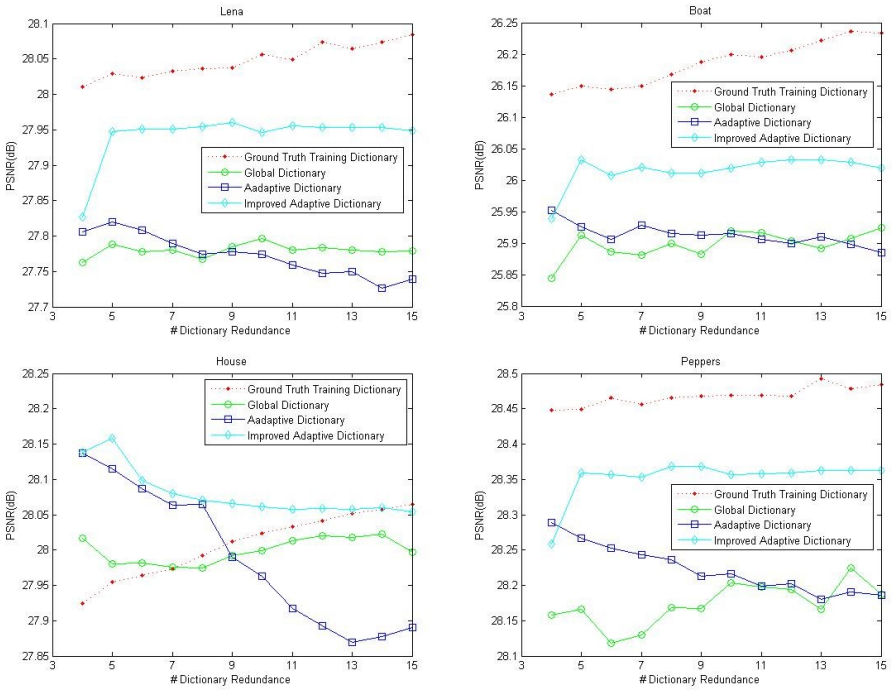


Fig. 5. The performance of denoising

noise image generated by deconvoluting the halftone image $z(n_1, n_2)$.

$$Y = P^{-1} z(n, n_2) \tag{19}$$

Kite et al. have verified that the gain block K is constant for a certain error diffusion filter in the linear model. We declare that the noise level is constant and we verify the noise level with enough experiments. Given a noise image without known the noise level, we use the algorithm to remove the noise with different noise level parameters. We assume the noise level is the one that can result in the best PSNR value, which are shown in Fig. 7. We can see that the noise level is $\sigma \approx 42$. The same as Jarvis where the noise level is $\sigma \approx 18$. For P and Q are the linear filter operators, so the noise level of $\gamma(n_1, n_2)$ in the error diffusion model is constant.

When the noise level is fixed in the halftone image, the denoising algorithm can be explored to directly remove the noise generated in the inverse halftoning processing. It provides a way for setting noise levels when using other denoising algorithms for image inverse halftoning. It is reasonable to resolving image inverse halftoning by using the image denoising algorithm.

6. Experimental results and analysis

In this section we first demonstrate that the proposed algorithm can remove noise more effectively. For well comparing with the adaptive dictionary learning denoising algorithm

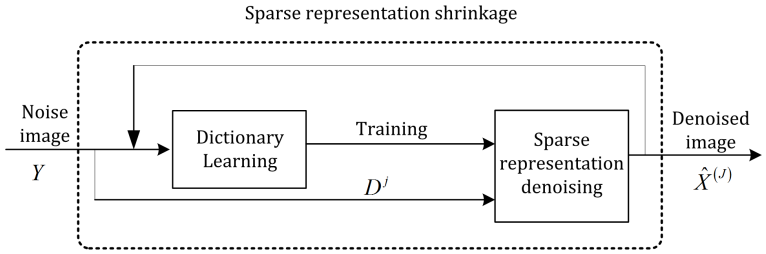


Fig. 6. Denoising by adaptive training dictionary iteratively

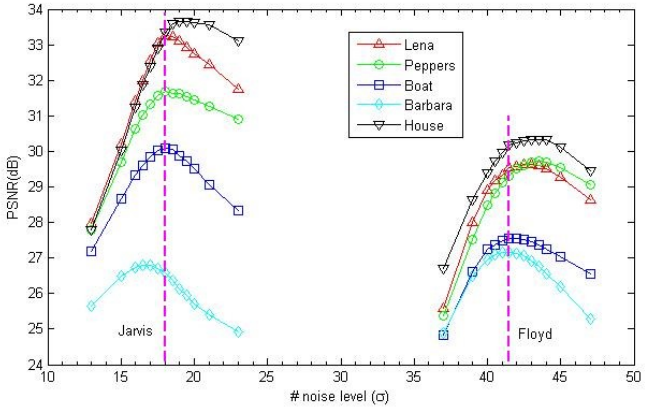
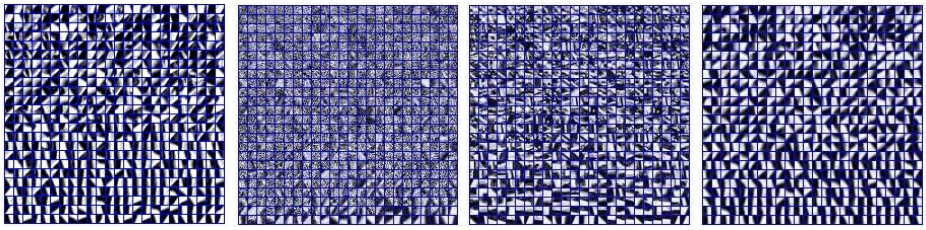


Fig. 7. The noise level of error diffusion model



(a) GT (b) AD (c) GTD (d) ID

Fig. 8. Different trained dictionaries

Algorithm 2 Image Denoising With Boosted Dictionary

Input: Giving image Y with addition Gauss white noise (the standard variance is σ).

Parameters: w is the number of iterations, J is the iteration number, and λ is the Lagrange operator.

1: Initialization: D^0 is the fixed number of samples randomly selected from the sample set after normalization. $Y^0 = Y$.

2: **for** $J = 1, \cdots, w$ **do**

3: Dividing the image $Y^{(J-1)}$ into patches as the training set $\{R_{ij}Y^{(J-1)}\}$. $D^{(J-1)}$ is formed by randomly selecting $(R_0 + J)$ sample. $D^{(J)}$ is got by training $D^{(J-1)}$. Set $X = Y^{(J-1)}$.

– *Sparse coding*: computing the sparse coefficients α_{ij} of the image patch in training set.

$$\hat{\alpha}_{ij} = \arg\min_{\alpha} \|D^{(J)}\alpha - R_{ij}X\|_2^2 + \mu\|\alpha\|_0 \tag{17}$$

– *Update the dictionary* D: suppose X is fix, in each time one of atoms in the dictionary and the corresponding correspond coefficient α_{ij} is updated.

– *Output* $D^{(J)}$

4: Sparse encoding: compute the coefficient α_{ij} of the noise image Y using Eq. (17).

5: Updated $D^{(J)}$ and the sparse encoding α_{ij} of noise image Y. The Jth denoised image can be computed.

$$\hat{X} = (\lambda I + \sum\nolimits_{ij} R_{ij}^T R_{ij})^{-1}(\lambda Y + \sum\nolimits_{ij} R_{ij}^T D^{(J)}\hat{\alpha}_{ij}) \tag{18}$$

6: $Y^{(J+1)} = \hat{X}$

7: **end for**

Output: $Y^{(J)}$

in the literature [5], zero-mean additive Gaussian white noise is used as the same as literature [5]. Then the proposed algorithm is applied to image inverse halftoning and compared with other benchmarks.

6.1. Experimental results of proposed method.

Literature [5] use the nature images without noise and the noise image itself to training the dictionaries respectively. The former one is named as global trained dictionary (GTD), and the other one is adaptive trained dictionary (AD). It as shown in Fig. 4, we select three different training images without noise to train the global dictionary. We name the dictionary trained by ground truth image as ground truth dictionary (GT), while the dictionary trained by our proposed method is called improved dictionary (ID). The dictionaries are shown in Fig. 8. It is obviously that the ID in our method is very similar with the GT objectively, while GTD is better than AD. At the same time the denoising performance of these dictionaries also show the same phenomenon subjectively, which is shown in Fig. 5 when the redundance is lower than 9. For convenience, we set the redundancy of all the dictionaries as 9, and the noise level is set as 50. All the algorithms are implemented in MATLAB 2015b on a 64bit Windows 10 Intel(R) Core (TM) i7 7500 CPU with 2.7 GHz and 8 GB RAM.

From the Fig. 8 we can see that the quality of ID is objectively better than the GTD and AD. The possible two reasons are declared as follows. In one hand the algorithm iteratively removes the noise of the image training set while the dictionary is the combination of the selected image patches in the initial training. Hence the dictionary quality will

Algorithm 3 Image Inverse halftoning Based on Sparse Representation With Boosted Dictionary

Input: Error diffusion halftone image $z(n_1, n_2)$.
Parameters: The error diffusion filter $h(n_1, n_2)$, Gain block K.

1: Step 1: Deconvolution
2: The estimated inverse halftone image with noise is obtained by using Fourier transform and inverse transform, which can be described as followed form using the Eq. (20).

$$Y = X + \nu \tag{20}$$

where the additive Gaussian white noise is $\nu = P^{-1}Q\gamma(n_1, n_2)$, and the ground truth image is $X = x(n_1, n_2)$, the inverse halftone image is $Y = y(n_1, n_2)$.
3: Step 2: Denoising based on sparse representation
4: Using the proposed algorithm 2 to adaptively training the dictionaries iteratively.

Output: $Y^{(J)}$ is the output inverse halftone image.

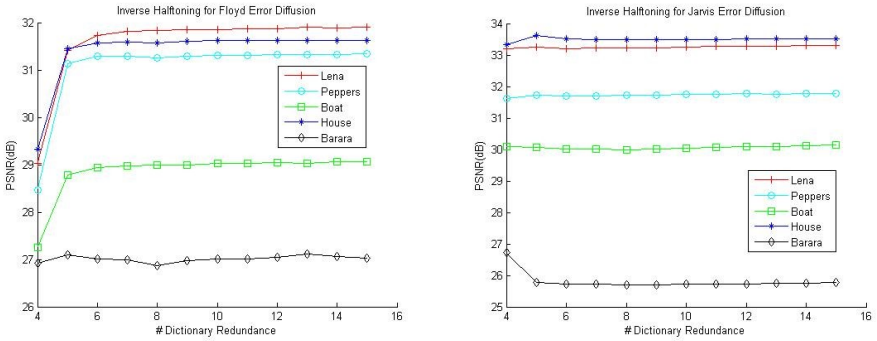


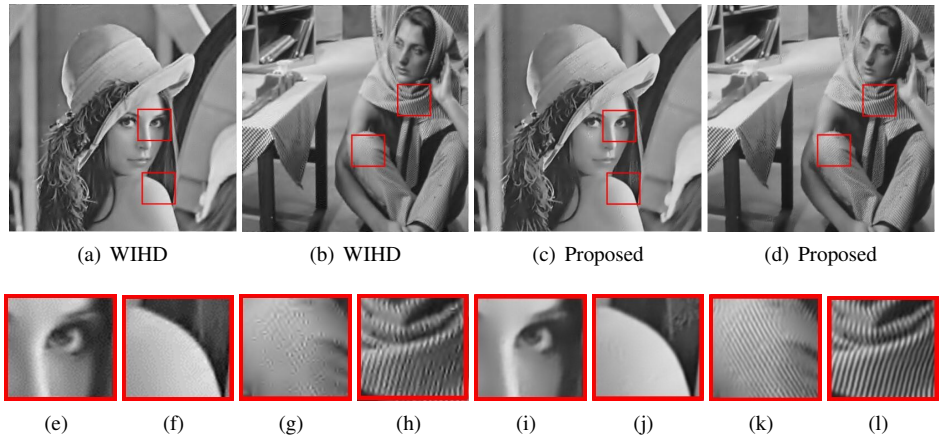
Fig. 9. The performance of proposed method

be boosted as the noise being removed iteratively. Then the performance of denoising is gradually improved. In the other hand, the redundancy of the dictionary will influence the denoising performance, so we increasing the redundancy in each iteration. Fig. 5 shows the denoising performance for several images with noise level $\sigma = 50$, which can be seen that the proposed algorithm can improve the denoising performance in the first several iterations and then tends to stable for *Lena*, *Peppers* and *Boat* images. For *House* image the performance of ID is better than the other three methods although the performance is not boosted with the iteration number increasing. Compared with the GTD, the proposed algorithm can improve the PSNR values of denoised images.

6.2. Experimental comparisons with benchmarks

Using the representative gray *Lena* and *Peppers* images with size 512×512, we compared the proposed algorithm with the benchmarks. We first halftoned the images with Floyd and Jarvis as the test halftone image, where we set the gain block K as 2.03 and 4.45 respectively. The deconvolution step is followed by the spare representation based denoising, and the PSNR results of the inverse halftone image are shown in Fig. 9 for

the halftone images, where the noise level σ is set to 40 and 18 respectively. The objective results are shown in Fig. 10, where the proposed method performance well than the benchmark. The details of the inverse halftoned image are zoomed in 5 times. The subjective results are shown in Tab. 1, where the texts of bigger PSNR values are bold. We can see that the PSNR values of our proposed method are almost better than WIHD's for all the test images especially for Jarvis error diffusion halftone images.



(a) WIHD (b) WIHD (c) Proposed (d) Proposed

(e) (f) (g) (h) (i) (j) (k) (l)

Fig. 10. Comparing with the benchmark.(e)-(h) are the details of WIHD (zoomed in 5 times);(i)-(l) are the details of proposed ones

For color image, we also compared out method with the method of sparse representation with double dictionaries learning [26], where the image size is $512 \times 512 \times 3$. In our method the redundance of our dictionary is 8. We used the Floyd error diffusion halftone image as the same as Son et. al. In order to compare with literature [26] fairly, the redundancy of our dictionary is set as 8 and the noise level is set as $\sigma = 40$ in our method. It can be seen that the proposed method performance better than the benchmark proposed by Son et. al. subjectively in R, G and B channels, which is shown in Tab. 2. In Fig. 11, the visual performance of our method is better than the method in literature [26] objectively, especially for the whole bear body in the result image.

Table 1. The PSNR(dB) values of image inverse halftone methods.

Error Filter	Images	WIHD	Proposed	Error Filter	WIHD	Proposed
Floyd	*Lena*	31.96	**32.01**	Jarvis	32.81	**33.39**
	Barbara	31.04	**31.39**		31.16	**35.86**
	Boat	25.71	**27.03**		25.12	**25.79**
	Hill	31.24	**31.62**		31.82	**33.53**
	House	**29.19**	29.06		29.78	**30.15**

There are more artificial and blur textures in the output continuous tone image from the methods in literature [26] and [22], which can be seen in Fig. 10 and Fig. 11, while our proposed model can reserve the texture details and structure information.

(a) Ground Truth (b) Halftone image (c) Method in [26] (d) Our method

Fig. 11. The objective results for benchmarks

Table 2. The subjective PSNR(dB) values of benchmarks.

Dictionary	64×256			64×512			64×256		
Channel	R-ch	G-ch	B-ch	R-ch	G-ch	B-ch	R-ch	G-ch	B-ch
[26]	25.10	25.21	25.28	25.16	25.26	25.32	25.28	25.34	25.40
Proposed	**26.29**	**26.38**	**25.96**	**26.83**	**26.69**	**26.33**	**26.82**	**26.71**	**26.35**

7. Conclusion

In this paper, we first proposed an image denoising method by boosting a learning dictionary iteratively, which efficient for the deconvoluted halftone image. Then we proposed an image inverse halftone approach for error diffusion halftone image by using the deconvolution and the denoising algorithm. Furthermore, we verified that the noise level of the halftone image for error diffusion is the constant, which gives the rationality for the inverse halftoning algorithm with deconvolution and denoising successively. Researchers can combination all kinds of deconvolution and denoising models (including the popular deep learning networks) to explore inverse halftoning approaches. Experiment results show that our algorithm is superior to other benchmarks. The limitation of our method is that it only inverses the error diffusion halftone images effectively. We would find a versatile inverse halftone method for both the dot dithering and error diffusion halftone image. The similar work about image descreening is also a challenge work should be considered in future.

Acknowledgments. This work was supported in part by the Zhejiang Public Welfare Technology Research Project Fund of China under Grant LGG22F020021 and LGG21F030013, and the City Public Welfare Technology Application Research Project of Jiaxing Science and Technology

Bureau of China under Grant 2021AY10071, 2020AY10009 and 2021AY10043. This work was also supported by the Dongguan Science and Technology of Social Development Program under Grant 20211800904512, Dongguan Sci-tech Commissoner Program under Grant 20231800500352, Characteristic Innovation Projects for Ordinary Universities in Guangdong Province under Grant 2021KTSCX301, Special projects in key areas of Department of Education of Guangdong Province under Grant 2022ZDZX1072.

References

1. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on signal processing 54(11), 4311–4322 (2006)
2. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM review 43(1), 129–159 (2001)
3. Damera-Venkata, N., Kite, T.D., Venkataraman, M., Evans, B.L.: Fast blind inverse halftoning. In: Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No. 98CB36269). vol. 2, pp. 64–68. IEEE (1998)
4. Elad, M.: Sparse and redundant representations: from theory to applications in signal and image processing, vol. 2. Springer (2010)
5. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image processing 15(12), 3736–3745 (2006)
6. Floyd, R.W., Steinberg, L.: An adaptive algorithm for spatial gray scale. Proceedings of the Society for Information Display 17 (1975)
7. Guo, J.M., Lee, H.: Watermarking in halftone images with mixed halftone techniques. International journal of imaging systems and technology 17(5), 303–314 (2007)
8. Hou, X., Qiu, G.: Image companding and inverse halftoning using deep convolutional neural networks. arXiv preprint arXiv:1707.00116 (2017)
9. Jarvis, J.F., Judice, C.N., Ninke, W.H.: A survey of techniques for the display of continuous tone pictures on bilevel displays. Computer graphics and image processing 5(1), 13–40 (1976)
10. Jimenez, F.P., Miyatake, M.N., Medina, K.T., Perez, G.S., Meana, H.P.: An inverse halftoning algorithms based on neural networks and atomic functions. IEEE Latin America Transactions 15(3), 488–495 (2017)
11. Kite, T., Evans, B., Bovik, A., Sculley, T.: Digital halftoning as 2-d delta-sigma modulation. In: Proceedings of International Conference on Image Processing. vol. 1, pp. 799–802. IEEE Comput. Soc (1997)
12. Kite, T.D., Evans, B.L., Bovik, A.C.: Modeling and quality assessment of halftoning by error diffusion. IEEE Transactions on Image Processing 9(5), 909–922 (2000)
13. Kuo, C., Rao, A.R., Thompson, G.: Wavelet based halftone segmentation and descreening filter. In: IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221). vol. 3, pp. 1573–1576. IEEE (2001)
14. Lee, H., Battle, A., Raina, R., Ng, A.: Efficient sparse coding algorithms. Advances in neural information processing systems 19 (2006)
15. Li, M., Zhang, E., Wang, Y., Duan, J., Jing, C.: Inverse halftoning methods based on deep learning and their evaluation metrics: A review. Applied Sciences 10(4), 1521 (2020)
16. Li, X.: Inverse halftoning with nonlocal regularization. In: 2011 18th IEEE International Conference on Image Processing. pp. 1717–1720. IEEE (2011)
17. Liu, J., Jumabayeva, A., Xu, Y., Wang, Y., Frank, T., Gat, S., Mor, O.B., Ben-Shoshan, Y., Ulichney, R., Allebach, J.: New results for aperiodic, clustered-dot halftoning. Electronic Imaging 2020(15), 195–1 (2020)

18. Liu, N., Pan, J.S., Sun, C., Chu, S.C.: An efficient surrogate-assisted quasi-affine transformation evolutionary algorithm for expensive optimization problems. Knowledge-Based Systems 209, 106418 (2020)
19. Lu, Z., Liu, G., Wang, S.: Sparse neighbor constrained co-clustering via category consistency learning. Knowledge-Based Systems 201-202, 105987 (2020)
20. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proceedings of the 26th annual international conference on machine learning. pp. 689–696 (2009)
21. Mairal, J., Elad, M., Sapiro, G.: Sparse learned representations for image restoration. In: Proc. of the 4th World Conf. of the Int. Assoc. for Statistical Computing (IASC). p. 118. Citeseer (2008)
22. Neelamani, R., Nowak, R.D., Baraniuk, R.G.: Winhd: Wavelet-based inverse halftoning via deconvolution. IEEE Transactions on Image Processing (2002)
23. Rubinstein, R., Peleg, T., Elad, M.: Analysis k-svd: A dictionary-learning algorithm for the analysis sparse model. IEEE Transactions on Signal Processing 61(3), 661–677 (2012)
24. Shao, L., Zhang, E., Li, M.: An efficient convolutional neural network model combined with attention mechanism for inverse halftoning. Electronics 10(13), 1574 (2021)
25. Siddiqui, H., Bouman, C.A.: Training-based descreening. IEEE transactions on image processing 16(3), 789–802 (2007)
26. Son, C.H.: Inverse halftoning based on sparse representation. Optics letters 37(12), 2352–2354 (2012)
27. Son, C.H.: Inverse halftoning through structure-aware deep convolutional neural networks. Signal Processing 173, 107591 (2020)
28. Son, C.H.: Layer decomposition learning based on gaussian convolution model and residual deblurring for inverse halftoning. Applied Sciences 11(15), 7006 (2021)
29. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Transactions on information theory 53(12), 4655–4666 (2007)
30. Wang, S., Wang, Z., Lim, K.L., Xiao, G., Guo, W.: Seeded random walk for multi-view semi-supervised classification. Knowledge-Based Systems 222, 107016 (2021)
31. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE transactions on pattern analysis and machine intelligence 31(2), 210–227 (2008)
32. Xia, M., Wong, T.T.: Deep inverse halftoning via progressively residual learning. In: Asian Conference on Computer Vision. pp. 523–539. Springer (2018)
33. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE transactions on image processing 19(11), 2861–2873 (2010)
34. Yen, Y.T., Cheng, C.C., Chiu, W.C.: Inverse halftone colorization: Making halftone prints color photos. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 1734–1738. IEEE (2021)
35. Yu, Z., Zheng, X., Huang, F., Guo, W., Sun, L., Yu, Z.: A framework based on sparse representation model for time series prediction in smart city. Frontiers of Computer Science 15(1), 1–13 (2021)
36. Zhang, H., Li, J.L., Liu, X.M., Dong, C.: Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection. Future Generation Computer Systems 122, 130–143 (2021)
37. Zhang, Y., Lu, Z., Wang, S.: Unsupervised feature selection via transformed auto-encoder. Knowledge-Based Systems 215, 106748 (2021)

Jun Yang received his M.S. degree from Guangdong University of Technology, Guangzhou, China, in 2010, and the Ph.D. degree from Sun Yat-sen University, Guangzhou, China, in

2016, both in Computer Science and Technology. From 2016, He joined the Jiaxing University at Jiaxing, China, where he was an assistant professor first and later associate professor. His current research interests include the areas of image processing, machine learning, artificial intelligence techniques and robotic control systems.

Zihao Liu received his Ph.D. degree in the School of Biosystems Engineering and Food Science, Zhejiang University, China, in 2017. From July 2019, he has joined the Jiaxing University where he is now an associate professor. His current research interests include image and video processing, Agricultural Mechanization Engineering and machine learning.

Li Chen received the Ph.D. degree in Harbin Institute of Technology. She is currently an associate professor in Jiaxing University. She is a member of China Computer Federation. Her main research interests include machine learning, and mobile communications technology.

Ying Wu is an associate professor in Jiaxing Nanhu University. She is pursuing the Ph.D. degree in Hangzhou Dianzi University. Her research interests are artificial intelligence and machine vision.

Gang Ke received the M.S. degree in computer application technology from Guangdong University of Technology, Guangzhou, China, in 2010. He is pursuing the Ph.D. degree in Macau University of Science and Technology. His current research interests include computer vision, deep learning, network security and IoT.

Received: August 08, 2022; Accepted: November 09, 2022.

Computer Science and Information Systems 20(2):831–856 https://doi.org/10.2298/CSIS220817023D

Selective Ensemble Learning Algorithm for Imbalanced Dataset

Hongle Du[1,2,3], Yan Zhang[1,3], Lin Zhang[1,3], and Yeh-Cheng Chen[4]

[1] School of Mathematics and Computer Application, Shangluo University,
726000, Shangluo, China,
{dhl5597,flyingzhang,zhlin002}@163.com
[2] University of the Cordilleras,
Aguio City, Philippines
[3] Shangluo Public Big Data Research Center,
726000, Shangluo, China
[4] Department of computer science, University of California,
Davis, CA, USA
ycch@ucdavis.edu

Abstract. Under the imbalanced dataset, the performance of the base-classifier, the computing method of weight of base-classifier and the selection method of the base-classifier have a great impact on the performance of the ensemble classifier. In order to solve above problem to improve the generalization performance of ensemble classifier, a selective ensemble learning algorithm based on under-sampling for imbalanced dataset is proposed. First, the proposed algorithm calculates the number K of under-sampling samples according to the relationship between class sample density. Then, we use the improved K-means clustering algorithm to under-sample the majority class samples and obtain K cluster centers. Then, all cluster centers (or the sample of the nearest cluster center) are regarded as new majority samples to construct a new balanced training subset combine with the minority class's samples. Repeat those processes to generate multiple training subsets and get multiple base-classifiers. However, with the increasing of iterations, the number of base-classifiers increase, and the similarity among the base-classifiers will also increase. Therefore, it is necessary to select some base-classifier with good classification performance and large difference for ensemble. In the stage of selecting base-classifiers, according to the difference and performance of base-classifiers, we use the idea of maximum correlation and minimum redundancy to select base-classifiers. In the ensemble stage, G-mean or F-mean is selected to evaluate the classification performance of base-classifier for imbalanced dataset. That is to say, it is selected to compute the weight of each base-classifier. And then the weighted voting method is used for ensemble. Finally, the simulation results on the artificial dataset, UCI dataset and KDDCUP dataset show that the algorithm has good generalization performance on imbalanced dataset, especially on the dataset with high imbalance degree.

Keywords: Under Sampling; Imbalanced Dataset; Clustering Algorithm; Selective Ensemble Learning.

1. Introduction

In practical application, due to the difficulty of collecting some samples or the high cost of collecting, the samples of some classes are less, which makes the sample size vary widely. In practical applications, such as network intrusion detection [1, 2], fault detection [3], medical diagnosis [4], etc., these datasets are called imbalanced dataset. The problem of imbalanced dataset in practical application is universal, while the traditional classification algorithms mostly focus on the balanced dataset. Therefore, under imbalanced dataset, the traditional classification algorithm will lead to over-fitting for the majority class and under-fitting for the minority class. That is to say, the classification accuracy of minority class is lower than that of majority class. However, in practical application, the classification accuracy of minority class is more important, such as the recognition rate of network intrusion behaviors, the recognition rate of diseases in medical diagnosis, etc.

Algorithms for imbalanced data are divided into two categories algorithm (algorithm-level methods and data-level methods). The data-level method mainly resamples the training dataset to make the resampled dataset reach balance, and it includes under-sampling for majority class [5-7], over-sampling for minority class [8-10] and combination of the above two methods [11]. Algorithm-level method mainly is to improve the algorithm to reduce the influence of imbalanced data, such as cost sensitive learning [12], single class classification [13], ensemble learning [14, 15], etc. The under-sampling algorithm is to discard some redundant samples from majority class samples according to a certain strategy, so that the number of samples of majority classes after resampling is equal to that of minority class. This makes the resampled dataset be the balanced dataset, such as Bootstrap method, clustering method, etc. Over-sampling is to generate some samples of the minority class according to a certain strategy, so that the number of two classes' samples is imbalanced. SMOTE algorithm [16] is the most representative oversampling algorithm. Because oversampling algorithm will increase the size of the original dataset and lead to over fitting phenomenon, this paper mainly discusses under-sampling algorithm. The purpose of under-sampling method is to make the number of two types of samples nearly equal after resampling, so what to discard and how many samples to discard are the key. The common under-sampling method is to select some majority class samples according to a certain strategy. The number of selected samples is equal to that of minority samples.

However, the essence of imbalanced data is the sample density imbalance of majority class and minority class [17]. Therefore, it is not accurate to resample simply according to the number of the minority class because the resampled dataset cannot be a real balanced dataset. It is more accurate to determine the number of resampled samples according to the relationship between class sample densities [1]. In addition, the classification performance of each base-classifier will also affect the performance of the final ensemble classifier. When sampling, it is necessary to keep the spatial distribution of the original samples well so that each base-classifier has better classification performance. In order to keep the spatial distribution of the original samples space, Liu [14] under-samples the samples from the majority class samples with the K-means clustering algorithm. In this way, the base-classifier has better classification performance, but how to determine K value is the key of this method. In addition, as the number of base classifiers increases, the similarity among them will also increase, which is not conducive to performance of the ensemble classifier. In order to improve

the classification performance of ensemble classifier, Zhou [18] selects better classification performance and larger difference degree of base classifiers. In order to improve the difference degree among base-classifiers, the K-means clustering algorithm is improved. Therefore, the proposed algorithm calculates the number K of under-sampling samples according to the relationship between class sample densities, and under-sampling uses the improved K-means clustering algorithm in this paper.

At the algorithm level method, ensemble learning algorithm is one of the important methods *t* to solve imbalanced data [4, 10]. EasyEnsemble algorithm and BalanceCascade algorithm [14] are the most representative algorithm to solve the classification problem of data imbalance with ensemble learning. In this algorithm, Bootstrap method is used to randomly under-sample from the majority class samples. Then combine the minority class and the under-sampled samples, and get a training subset. Multiple training subsets are trained and multiple base-classifiers can be obtained by repeating the above processes many times. Then ensemble the multiple base-classifiers and obtain the ensemble classifier. The Bootstrap method is under-sampling with playback, which lead to losing of some sample information. In order to let the classifier, learn enough sample information, we may increase the number of resampling times.

However, with the increase of resampling times and the number of base-classifiers, the more similar among the base-classifiers are, the difference degree among multiple base-classifiers will reduce. This will lead to reduce the generalization performance of the final ensemble classifier. To reduce this impact, Zhou [14] prove that select some base-classifiers with good performance and big difference for ensemble has better generalization performance than all base-classifiers, and proposed the selective ensemble algorithm of GASEN. The algorithm and its improved algorithm [4, 19, 20] are to select some base-classifiers with good performance according to certain methods for ensemble. The problem of selection ensemble learning is how to judge the quality of base-classifiers. Potharaju [21] points out that the base-classifier with larger difference degree and better classification performance is a good classifier, which is suitable for ensemble. Subsequently, many scholars [22, 23] improved the selective ensemble learning algorithm, which are mainly focused on how to select the base-classifiers and which base-classifiers to be selected. Generally speaking, it is to select good base-classifiers for integration, so the problem is transformed into how to evaluate the advantages and disadvantages of the base-classifier. In order to describe the generalization performance of the base-classifier, the generalization error is often used to evaluate the base-classifier. It is mostly based on the classification performance and the difference degree among the all base-classifiers.

However, the existing evaluation methods are all aimed at the balanced dataset. For the imbalanced data, the calculation method of the difference degree and the classification performance of the base-classifier are different from those of the balanced dataset. This paper presents an evaluation method of base-classifiers for unbalanced dataset. Then the base-classifier is evaluated and selected according to the evaluation results.

In view of the above problems, an improved selective ensemble learning algorithm for imbalanced data based on the clustering under-sampling (CSEL for short) is proposed in this paper. According to the idea of EasyEnsemble algorithm, the algorithm is improved from the calculation of the number of under-sampling samples, the method of under-sampling, the evaluation method of the base classifier and the calculation

method of base-classifier weight. First, the proposed algorithm calculates the number of under-sampled samples according to the ratio between the two classes' densities. Then, we under-sample samples from majority class with the improved k-means clustering algorithm. Then, all cluster centers (or the sample of the nearest cluster center) are looked as the new majority class samples, and obtain a balanced training subset by combining with the under-sampled samples and the minority class's samples. Repeat above process and generate multiple base-classifiers. However, increasing the number of base classifiers will lead to reduce the difference degree among base-classifiers. Therefore, it is necessary to select some base-classifier with good classification performance and large difference for integration. In the stage of selecting base-classifiers, according to the difference and performance of base-classifiers, we use the idea of maximum correlation and minimum redundancy to select base-classifiers. In the ensemble stage, G-mean or F-mean that is often used to measure the classification performance of classifier for imbalanced data is used to calculate the base-classifier weight in this paper. Then obtain the ensemble classifier with the weighted voting method. The effectiveness of the proposed algorithm is proved by simulation experiment on artificial dataset, UCI dataset and KDDCUP dataset. The experiment mainly compares the experimental results of various algorithms from accuracy, G-mean, F-mean, AUC, Recall, Precision and ROC curve. The experimental results show that the proposed algorithm in this paper has a good classification performance under imbalanced dataset.

2. Related Work

2.1. EasyEnsemble algorithm

Easyensemble algorithm is the under-sampling algorithm. It randomly selects some samples that the number is equal to the number of the minority class from majority class samples. Put the minority class samples and under-sampled samples together to obtain a balanced data subset [14, 24]. Then, train on the imbalanced training subset with the AdaBoost algorithm [24] and get base-classifier. We can get multiple balanced training subsets and multiple base-classifiers by repeating the above process many times. Finally, ensemble classifier can be getting through by integrating the multiple base-classifiers according to a certain strategy.

Suppose that P presents the minority class and N presents the majority class, the number of minority class samples is $|P|$ presents the number of minority class samples and $|N|$ presents the number of majority class samples. The number of iterations of AdaBoost algorithm is S_i. The detailed describe of EasyEnsemble algorithm can be shown as follows:

Algorithm1: EasyEnsemble

1.for i=1:T

1.1 From majority class samples, randomly take $|P|$ samples with the Bootstrap method.

1.2 Put the minority class samples and selected samples together and get the training subset. With the AdaBoost algorithm, train on training subset to obtain the base classifier. The obtained base-classifier can be represented by the following formula:

$$H_i(x) = \text{sgn}(\sum_{j=1}^{s_i} a_{ij} h_{ij}(x) - \theta_i)$$

2. Then the final classifier is represented by the formula:

$$H(x) = \text{sgn}(\sum_{i=1}^{T} \sum_{j=1}^{s_i} a_{ij} h_{ij}(x) - \sum_{i=1}^{T} \theta_i)$$

From the above algorithm process, we can see that the EasyEnsemble algorithm is a data-level method. During under-sampling, in order to reduce the loss of sample space information, multiple times resampling is used for multiple iterations. The obtained base-classifiers are integrated to obtain the ensemble classifier. There are three shortcomings in this algorithm: (1) the essence of dataset imbalance is the sample density imbalance of majority class and minority class. (2) Increase the number of iteration times will also reduce the difference degree among the base-classifiers, which will affect the classification performance of the ensemble classifier [12]. Therefore, it is better to select some base-classifiers with high accuracy and great difference to integrate than all base-classifiers [12]. That is to say, we should adopt selective ensemble learning algorithm. (3) The generalization errors of the base-classifiers are different, especially in the case of imbalanced dataset, so weighted voting integration is more conducive to improving the performance of the ensemble classifiers. This paper also improves the algorithm from these three aspects.

2.2. The Influence of Class Sample Density on Classifier

Data imbalance means that the number of one class sample is far more than that of other classes' sample. The class that the number of samples is more is called the majority class; on the contrary, it is called minority class. The imbalanced dataset has great influence on the final classification, and detailed information can be found in relevant literature, such as [1, 9, 24]. In this section, we mainly discuss the influence of resampling sample number on ensemble classifier performance. In order to verify the influence of class sample density on classifier, the Hyperplane graph of classifier under linear separable and linear non separable dataset is given in Fig. 1 and Fig. 2 respectively. In the Fig.1 (a), the majority class with 200 samples fellow the $U([0,1]\times[0,1])$ distribution. The minority class with 50 samples fellow the $U([0,1]\times[1,2])$ distribution. In Fig.1 (b), the majority class samples fellow the $U([0.8,1]\times[0,1])$ distribution and the number of samples is 50. The minority class samples fellow $U([0,1]\times[1,2])$ and the number of samples is 20. Fig.1 shows that the decision Hyperplane with the SVM algorithm. Left fig.1 (a) is the classification Hyperplane with the sample number ratio of 100:20. The right fig. (b) is the classification Hyperplane with the sample number ratio of 1:1, but the space area of the two classes is different. From Fig.1 (b), we can see that under-sampled samples still are the imbalanced dataset if the numbers of samples of two classes are equal after resampling. That is to say, the sample density of the two classes is different in fig. 1(a). From the Fig 1, we can see that the classification Hyperplanes drifts towards the class with small class density.

Fig.2 shows uniformly distributed artificially generated samples. The + sign represents ten samples of uniformly generated in a circle with a radius of 1, and the X sign represents one thousand samples of uniformly distributed in a circle with a radius of 1 to 2. The classification Hyperplane constructed by SVM algorithm is shown in Fig.1. Among them, the left figure is a classification Hyperplane based on the original dataset. The right figure is a Hyperplane constructed by support vector machine after K-means clustering of majority class samples (k = 10, i.e., taking samples with the same number of minority class samples). As can be seen from Fig. 1, after under-sampling, the classification Hyperplane is shifted to direction of the original majority class. This is due to a new imbalance in the dataset after under-sampling.

Fig.1 and Fig.2 show that the essence of imbalanced dataset is not the imbalance of samples number, but the imbalance of class sample density. Wu [10] also proves that the essence of imbalanced dataset is the imbalanced density of class samples. Wang [11], *K* is set as the number of minority class samples to carry out *K-means* cluster algorithm, then all cluster centers (or the sample of the nearest cluster center) are looked as the new majority class samples. That is to say, it is under-sampling for majority class samples. This method may lead to new imbalanced dataset after under-sampling. In order to make the resampled data set approximate to the balanced dataset, the number of under-sampled samples is determined according to the ratio between the two class samples densities in this paper.

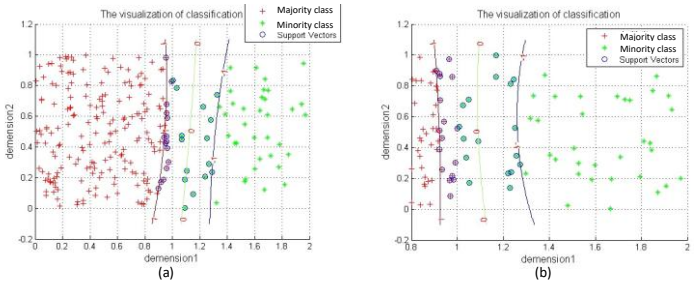


Fig.1 The classification Hyperplane of different density on linear separable dataset

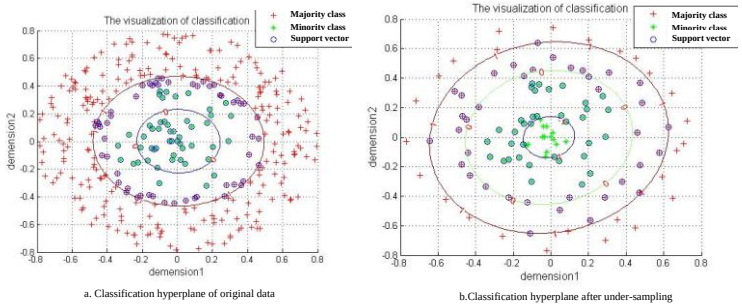


Fig.2 The classification Hyperplane of different density on nonlinear separable dataset

2.3. Under-sampling Method

EasyEnsemble algorithm [14] uses bootstrap method to select the same number of minority class sample from the majority class, and then get one training subset. Random under-sampling can ensure that there is a larger difference between the multiple base-classifiers. At the same time, the performance of some base-classifiers is poor performance because the selected samples cannot keep the spatial distribution of the original sample space. This will reduce the classification performance of the ensemble classifier. In reference [11], in order to ensure that the spatial distribution of the selected samples is similar to that of the original samples, majority class samples are under sampled with the *K-means* clustering algorithm. Then *K* clusters are obtained and the *K* centers of each cluster are looked as the new sample. In order to more accurately maintain the spatial distribution of the original samples after under-sampling, the sample closest to the cluster center is regarded as the new selected sample [20]. Because the cluster center of this method hardly changes at each iteration, the difference of resampled dataset is relatively small, which results in the small difference among the base-classifiers. The improved under-sampling methods [11],[20] are used to keep the spatial distribution characteristics of the original sample space, and make the base-classifier have better classification performance. However, it will also result in have higher similarity among the base-classifiers [25],[26], which is not suitable for ensemble learning. Therefore, this paper improves the *K-means* clustering algorithm with the idea of density cluster method into the *K-means* clustering method to make, and then uses the improved *K-means* clustering algorithm to under-sampling the majority class samples. It can not only keep the original spatial distribution of the samples after under-sampling, but also improve the difference among the datasets obtained by resampling.

2.4. Selection Method of Base-Classifier

Increasing the number of base-classifiers will lead to reduce the difference degree among the base-classifiers. It will reduce the classification performance of the final ensemble classifier. So, it is a good choice to select the base-classifier with batter classification performance and larger difference degree for ensemble [12]. The key of selective ensemble learning algorithm is how many and which base-classifiers are selected. Therefore, it is necessary to define the evaluation indictor of base-classifier to evaluate the classification performance as the criteria for selecting the base-classifier. It has been proved in reference [15] that the greater difference degree of the base-classifiers, the better the performance of the ensemble classifier. Almost all of the selection methods of base-classifier are based on the generalization errors of base-classifier. And the generalization error of base-classifier is evaluated with the accuracy of base-classifier and the difference degree among base-classifiers [4, 20]. In reference [19], ten methods to measure the difference of base-classifiers are given.

Almost all methods calculate the evaluation indictor according to the relationship between the test results of any two base-classifiers on the same dataset. That is, the inconsistency among the classification results of the same dataset by the different base-classifier. In reference [4], information entropy is used to calculate the difference degree

among the multiple base-classifiers. In reference [20], the difference degree of each base-classifier is calculated by the mutual information and classification error between any two base-classifiers. At present, the commonly used classifier difference measurement methods include Q statistics, correlation coefficient, double error, inconsistent measurement, etc. Suppose there are two classifiers f_i and f_j . The number of samples that are correctly classified by two base-classifiers is presented as N^{11}, and the number of samples that are incorrectly classified by two base-classifiers is expressed as N^{00}. and the number of samples that are correctly classified by f_i and incorrectly classified by f_j is expressed as N^{10}, and the number of samples that are incorrectly classified by f_i and that are correctly classified by f_j is expressed as N^{01}, and the number of all samples is expressed as N. Then the inconsistent measure between base-classifiers can be expressed as:

$$D_{ij} = \frac{N^{01} + N^{10}}{N} \tag{1}$$

Then the inconsistency between the classifier and the classification target can be expressed as:

$$D(f_i, L) = \frac{N^{01} + N^{10}}{N} \tag{2}$$

Where L is the classification target, and N^{10}, N^{01} and N have the same means as above.

3. Algorithm Description

3.1. Determine the number of under-sampled samples

Suppose the majority class is expressed as T_{maj}, and the minority class is expressed as T_{min}. The average distance between every two samples in the majority class is expressed as AD_{maj}, The average distance between every two samples in the minority class is expressed as AD_{min}, The number of the majority class sample is expressed as N_{maj} and the number of the minority class sample is expressed as N_{min}. Then, AD_{maj} and AD_{min} can be calculated by formula (3) and formula (4).

$$AD_{maj} = \frac{1}{N_{maj}} \sum_{i=1}^{N_{maj}} \sum_{j=i+1}^{N_{maj}} d(x_i, x_j) \tag{3}$$

$$AD_{\min} = \frac{1}{N_{\min}} \sum_{i=1}^{N_{\min}} \sum_{j=i+1}^{N_{\min}} d(x_i, x_j) \tag{4}$$

Where, $d(x_i, x_j)$ represents the distance between two samples x_i and x_j. $d(x_i, x_j)$ can be calculated by Manhattan distance, European distance and so on. In this paper, European distance is used.

Class sample density is the number of samples per unit volume. However, the volume of a class is easily affected by noise data. In order to simplify the calculation, the average distance between samples in one class is used to calculate the class sample density. The volume of the class sample can be expressed as:

$$V_{maj} = AD_{maj} * N_{maj} \quad (5)$$

$$V_{\min} = AD_{\min} * N_{\min} \quad (6)$$

The number of selected samples is proportional to the class volume, so the number of under-sampled samples for majority classes can be expressed as:

$$SN = \left\lceil \frac{V_{maj}}{V_{\min}} N_{\min} \right\rceil \quad (7)$$

3.2. How to Select Samples

K-means clustering algorithm determines which cluster to join according to the distance from sample to each cluster center. In order to reflect the distribution of class samples density, this paper presents an improved k-means clustering algorithm. The improved algorithm selects the initial cluster centers according to the density distribution of samples. "Sample density" refers to the number of samples in a specified neighborhood centered on the sample, which is called the "density" of the sample. The algorithm first calculates the "sample density" of each sample, and then sorts it according to the sample density. Then randomly select some samples with high density as the initial cluster centers. On the one hand, it can keep the spatial distribution of samples; on the other hand, it can avoid the influence of noise sample. Then, *K* samples that are not adjacent to each other are randomly selected from the selected samples as the initial cluster center. The detailed pseudo-code of algorithm is shown as follows:

Algorithm 2: improved algorithm of *K*-means clustering algorithm

Input: dataset is T $(x_1, x_2, \mathrm{L}\ , x_n)$ with *n* samples, the neighborhood radius is *R*

Output: the center of each cluster (or the sample closest to the cluster center)

Step 1: The density of each sample x_i in *T* and the samples in its neighborhood are calculated. Then, they were arranged in descending order by the sample density, and select a certain proportion of samples and put them into *S* (In the experiment of this paper, the first 80% of samples are selected);

Step 2: For i=1: *K*

Random sample C_i is selected from *S*, and delete C_i and the samples in its neighborhood from *S*;

Step 3: Calculate the distance from sample to each cluster center, and put the sample into the cluster closest from the sample;

Step 4: Calculate the coordinate mean value of all samples in each cluster, and let this mean value to be the new cluster;

Step 5: Repeat the above process until the cluster center does not change in a large range or the clustering times meet the requirements.

The improvement of *K-means* algorithm is mainly on the selection method of initial class center. The neighborhood radius has a great influence on the algorithm. If the domain radius is too large, it may lead to the selection of not enough *K* initial samples; if the domain radius is too small, each sample has only one neighborhood, and the improved algorithm is the original *K*-means algorithm. In this paper, we choose neighborhood radius by experience, and use two times of the average distance of class samples as the neighborhood radius.

3.3. Selection Method of Base-classifier

The generalization performance and difference of the base-classifier will affect the generalization performance of ensemble classifier [12]. The generalization error is expressed by the formula: $E = \overline{E} - \overline{D}$.

$\overline{E}$ is the average of generalization errors of all base classifiers and $\overline{D}$ is the average of difference degree of the all base-classifiers. According to the above formula, reducing the generalization error or (and) increasing the difference degree can reduce the generalization error of the ensemble classifier. Therefore, we can define the evaluation method of the base-classifier from these two aspects, then evaluate the base-classifier, and select the base-classifier according to the evaluation results in this paper.

Selective ensemble learning is to discard the classifiers that have poor classification accuracy and high redundancy. Maximum correlation minimum redundancy criterion [27] is used to select features according to the correlation between the features of dataset and classification objectives. Selecting the features and selecting classifier have similar objectives and basis. Therefore, the criterion can be applied to select the base-classifiers of high generalization error. Given f_i represent the classification results of the *i*-th base-classifier and *L* represent the classification target. Then similarity degree between the base-classifier and the classification target can be expressed as the formula (8). It is the average mutual information among the classification results and the classification target.

$$E(S, L) = \frac{1}{|S|} \sum_{f_i \in S} I(f_i; L) \tag{8}$$

Redundancy between classifiers can be measured by the average of mutual information among the all base-classifiers. The calculation method can be expressed as the formula (9).

$$D = \frac{1}{|S|^2} \sum_{f_i \in S} \sum_{f_j \in S} I(f_i; f_j) \tag{9}$$

Then the maximum correlation minimum redundancy criterion for classifier selection can be expressed as:

$$GE = \max \phi(E, D) \qquad (10)$$

From the above calculation method, we can see $I(f_i; f_j)$ focus on the probability of occurrence of each class. Then, the similarity between the two classifiers is calculated according to the probability, not whether the classification results are the same for each sample. Therefore, this method is not enough accurate to describe the difference between two classifiers, especially in the case of imbalanced dataset. Therefore, in this paper, the correlation between classifier and classification target is defined as:

$$E_i = 1 - D(f_i, L) \qquad (11)$$

The difference D_{ij} between base-classifiers is calculated by formula (1), then the redundancy between the base-classifier and other base-classifiers is expressed as:

$$D_i = 1 - \frac{1}{|S| - 1} \sum_{f_j \in S} D_{ij} \qquad (12)$$

By formulas (11) and (12), the generalization performance of the base-classifier is defined as:

$$GE_i = E_i - D_i \qquad (13)$$

Or

$$GE_i = E_i / D_i \qquad (14)$$

3.4. Ensemble method of base-classifier

The ensemble method of the base-classifiers mostly uses voting method and average method. Voting method also includes hard voting and weighted voting. Hard voting is to predict which class has more classifiers and which class is the final one. Weighted voting method obtains the ensemble classifier based on the weight of base-classifier that is calculated according to a certain calculation method (for example, accuracy, precision, recall, etc.). For imbalanced dataset, more emphasis is on the classification accuracy of minority class samples. Calculating the weight with accuracy cannot accurately reflect the importance of the base-classifier. For imbalanced dataset, *F-mean* and *G-mean* are often used to evaluate the classification performance of classifier. These two evaluation indications are calculated with the confusion matrix [28]. Table 1 shows the confusion matrix for the binary classification problems.

Table 1. Confusion matrix of binary classification

class		Prediction class	
		Positive	Negative
Real class	Positive	TP	FN
	Negative	FP	TN

According to the confusion matrix, the evaluation indication given above can be expressed as follow formulas:

$$\mathrm{Re}call = \frac{TP}{TP + FN} \quad (15)$$

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP} \quad (16)$$

$$F - mean = \frac{2 \times \mathrm{Pr}ecision \times \mathrm{Re}call}{\mathrm{Pr}ecision + \mathrm{Re}call} \quad (17)$$

$$G - mean = \sqrt{\frac{TN}{TN + FP} \times \frac{TP}{TP + FN}} \quad (18)$$

Formula (17) shows that *F-mean* takes into account the precision and recall. Only when the two values are larger, *F-mean* is larger. Therefore, under imbalanced dataset, this indicator can well describe the classification performance of classifier. Formula (18) shows that *G-mean* takes into account the classification accuracy of two classes. Only when the classification accuracy of the two classes is high, the value of *G-mean* will be larger. When the accuracy of any class is low, the value of *G-mean* is low. In this paper, we adopt these two evaluation indicators to calculate the weight of classifier. In order to accurately describe the importance of each base-classifier in the ensemble classifier, the corresponding weights of *F-mean* or *G-mean* are normalized as follows:

$$W_i = \frac{G_i}{\sum_{i=1}^{K} G_i} \quad (19)$$

$$W_i = \frac{F_i}{\sum_{i=1}^{K} F_i} \quad (20)$$

The ensemble classifier can be expressed as the formula (21).

$$H = \arg\max_{y \in Y} \sum_{i=1}^{m} W_i h_i \quad (19)$$

3.5. Algorithm Description

Fig. 3 shows the algorithm flow chart. Combining with the number of under-samplings, and the method of under-sampling, and the selection of base-classifier, and the ensemble method of multiple base-classifiers and so on, an improved selection ensemble learning algorithm based on under-sampling for imbalanced dataset is proposed in this paper. The proposed algorithm is mainly divided into three stages: data processing, construction base-classifier, selecting base-classifier and obtaining the ensemble classifier. First of all, the imbalance degree between any two base-classifiers is calculated according to the class sample density, and K is determined according to the imbalance degree of two class samples. Then, the improved *K-means* clustering method for under-sampling is given, and then it is used to under-sample from majority samples. Combine with the minority class samples and under-sampled samples to get the training

subset. Then obtain multiple base-classifier according to someone machine learning algorithm on training subset.

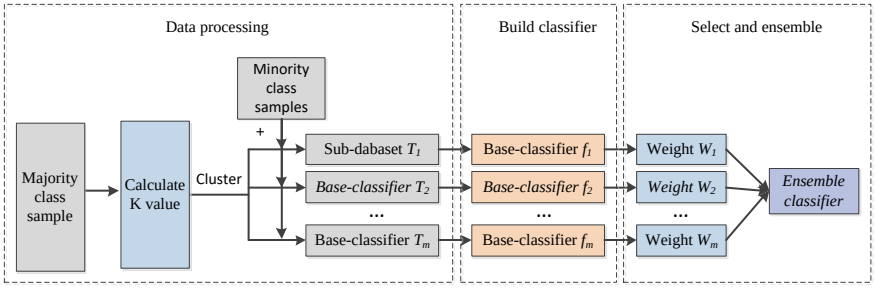


Fig. 3. flow of the algorithm

Then, according to the mutual information, we define the measurement method of the difference degree among the base-classifiers. According to the difference degree and the classification performance, better base-classifiers are selected. Finally, the weight of each base-classifier for the imbalanced data is calculated according to the *G-means*, and then, the multiple base-classifiers are integrated according to the weight. The three stages (data processing, construction base-classifier, selecting base-classifier and obtaining the ensemble classifier) of the algorithm are all designed for imbalance dataset. Each stage is designed to reduce the impact of imbalance dataset on base-classifier and ensemble classifier. The detailed pseudo-code of algorithm is shown as follows:

Algorithm 3: Selective ensemble learning algorithm based on clustering under-sampling for imbalanced dataset

Input: training dataset *TD*, number of algorithm iterations *T*, number of selection base-classifiers *m*, *K* value in *k*-means algorithm

Output: Ensemble classifier

Step 1: Training dataset *TD* is delimited majority class sample set $T_+$and minority class sample set T_-. $|T_+|$and $|T_-|$is the number of samples in the dataset T_+ and T_-;

Step 2: Ccalculate the class sample density and get the number of cluster *SN* using the formula (5);

Step 3: for i=1:*T*

3.1 According to algorithm 2, select the *SN* samples from majority class T_+ as T_+', and then union of two sets T_+' and T_- is the training subsets $SubT_i$;

3.2 Train and obtain the base-classifier h_i, and use h_i to predict the training dataset *TD*. Then the weight W_i of the base-classifier is calculated by formula (18);

End for i

Step 4: Formula (11) and (12) are used to calculate the difference degree and classification performance between base-classifiers, and then the generalization performance GE_i of every base-classifier is calculated with formula (13), and sort by GE_i;

Step 5: Select *M* base-classifiers with small generalization error, and ensemble according to formula (21) to get the weighted ensemble classifier

$$H = \arg\max_{y \in Y} \sum_{i=1}^{m} W_i h_i$$

The purpose of resampling is to make the resampled dataset reach balance, which can improve the performance of base-classifier under imbalanced data. It makes the base-classifier have better classification performance. It is more accurate to calculate the number of under-sampling samples according to the class sample density. Then under sampling adopt improve *K-mean* clustering algorithm. Compared with the direct use the number of minority class samples, classification performance of the base-classifier using this method is better.

According to the class sample density, calculate the selected the number of samples and obtain the *K* clusters using the improved *K-means* cluster algorithm. The obtained clustering center can keep the spatial distribution of the original data as much as possible. In addition, it can avoid the impact of noise samples, and further improve the classification performance of the base-classifier. Compared with calculating the weight with accuracy, calculating the weight with *G-mean* is more accurate to describe the classification performance for the imbalanced dataset.

When increase the number of iterations and obtain the more base-classifiers, the similarity among the obtained base-classifiers will increase and the generalization performance of the final ensemble classifier will reduce. Therefore, it is necessary to select the base-classifiers with high accuracy and large difference from the total base-classifiers to integrate. The accuracy and difference degree are expressed as the generalization error of the base-classifier. Formulas (13) and (14) give the calculation method of generalization error under imbalanced dataset. According to the calculation of generalization error, select several base-classifiers with small generalization error for ensemble. The selection number of the base-classifier will also impact on the generalization performance of the final ensemble classifier. In the experimental part of this paper, the effect of selection proportion on the performance of the ensemble classifier is given through experiment method. The conclusion is that the selected proportion is related to the specific dataset. For the selected base classifier, weight is calculated according to formula (19) or (20) and weighted voting is used to integrate. Such weights can more accurately reflect the classification performance of the base-classifier. Finally, the final ensemble classifier is obtained according to formula (20). From the whole process of the algorithm, we can see that every step of the algorithm is designed to reduce the influence of imbalanced dataset on the performance of ensemble classifier.

4. Experimental Results and Analysis

Because the algorithm has certain randomness, the experimental results of every algorithm in this paper are the standard deviation and mean of 10 experiments. In the experiment part of this paper, all the results are from the simulation experiment on MATLAB 2015b. This section mainly verifies the effectiveness of the proposed algorithm. The experimental results include three parts: the artificial dataset, UCI dataset and KDDCUP dataset. Formulas (13) and (14) have an impact on the performance of the algorithm, as do formulas (19) and (20). In the experiment, we also compare the experimental results of different formulas.

4.1. Experimental Results on Artificial Dataset

In this section, the effectiveness of the algorithm is verified by simulation experiments on artificial dataset. The artificial dataset includes two kinds: linear separable dataset and linear inseparable dataset. Randomly generate two classes' imbalanced samples of uniform distribution to verify the classification performance of the algorithm under imbalanced dataset. The first-class samples (majority class) are $\boldsymbol{U}([0,1.1]\times[0,1])$, and the second-class samples (minority class) are $\boldsymbol{U}([0.9,2]\times[0,1])$. The majority class includes 600 samples, and the minority class includes 30 samples. The test dataset also uses uniform distributed artificial data, but it is balance dataset. The first type of sample is $\boldsymbol{U}([0,1.1]\times[0,1])$, and the second-class sample is $\boldsymbol{U}([0.9,2]\times[0,1])$. There are 500 samples for each class.

From the above process of building dataset, we can see that the two classes of samples overlap in space, which increases the difficulty of classification. The mean and standard deviation of ten experiments of the artificial linear separable dataset is shown in table 2. The ROC curve of four algorithms is shown in Fig. 3, Moreover, the ROC curve of four algorithms is the one random experiment result. Table 2 and fig. 4 mainly compare the experimental results of BalanceCascade algorithm, EasyEnsemble algorithm, AdaBoost algorithm and CSEL algorithm. Table 2 shows the detailed experimental results. The change curve of each evaluate indicator of ten experiments is shown in Fig. 5. From the Fig. 5, we can see that *AUC* and *G-means* vary greatly and the *F-means*, *Precision* and *Recall* changes smoothly from Fig.5.

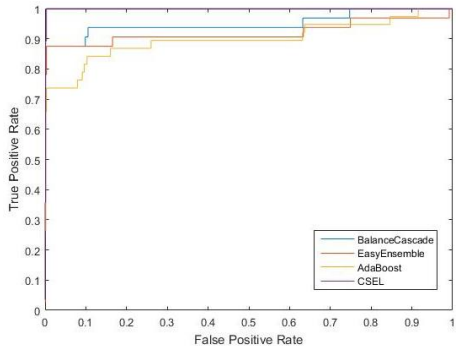


Fig. 4. ROC curve on linear separable dataset

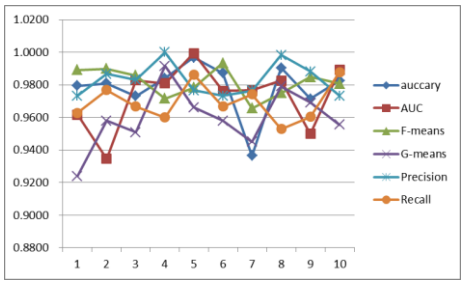


Fig. 5. Change curve of ten experiments on linear separable dataset

Table 2. Comparison of four algorithms on linear separable dataset

Algorithm	Evaluation Indicators (mean ±standard deviation)					
	Accuracy	AUC	F_mean	G_mean	Recall	Precision
CSEL	97.68±2.03	97.64±1.91	97.51±1.87	96.85±2.33	98.74±0.96	97.64±1.28
BalanceCascade	96.73±3.11	94.94±4.06	96.89±2.86	95.47±3.27	96.49±2.83	95.48±3.16
EasyEnsemble	92.45±2.45	92.95±3.44	95.41±2.14	93.79±2.88	94.13±3.08	93.97±3.67
Adaboost	92.38	92.42	93.11	92.86	92.58	92.84

In order to verify the difference between formula (13) and formula (14), ten experiments were carried out with formula (13) and formula (14) respectively. The standard deviation and mean of ten experiments about four algorithms (BalanceCascade algorithm, EasyEnsemble algorithm, AdaBoost algorithm and CSEL algorithm) is shown in table 3. From experimental results, we can see that the classification performance with formula (13) is better than formula (14).

Table 3. Experimental results with formula (13) and formula (14)

Formula	Evaluation Indicators (mean±standard deviation)					
	Accuracy	AUC	F_mean	G_mean	Recall	Precision
Formula (13)	97.68±2.03	97.64±1.91	97.51±1.87	96.85±2.33	98.74±0.96	97.64±1.28
Formula (14)	96.14±3.11	95.83±2.54	95.61±2.86	94.97±2.85	96.88±1.47	95.51±1.79

This part is an experiment on the linear inseparable data, which uses two classes of concentric circle samples of $\begin{cases} x = \rho g \cos\theta \\ y = \rho g \sin\theta \end{cases}, \theta \in U[0, 2\pi]$. The first-class sample (the majority class) is a uniform distribution with radius $\rho \in [0,1.1]$, and the second-class sample (the minority class) is a uniform distribution with radius $\rho \in [0.9, 2]$. The majority class includes 1000 samples, and the minority class includes 100 samples. The two classes of test dataset all include 500 samples, and the distribution is the same as the training dataset. The detailed results of ten random experiments are shown in table 4 about evaluate dictators of *Accuracy*, *AUC*, *F-means*, *G-means*, *Recall* and *Precision*, and the ROC curves about four algorithms are random one experiment is shown in Fig. 6. The change curve of each evaluate indicator of ten experiments is shown in Fig.7. From the fig.7, we can see that *AUC* and *G-means* vary greatly and the *F-means* from Fig.7.

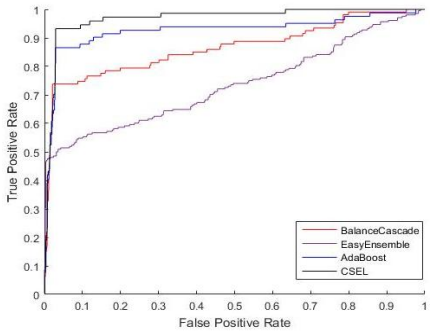


Fig.6. ROC curve on linear inseparable dataset

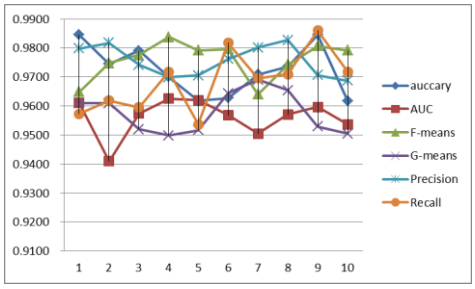


Fig. 7. Change curve of ten experiments on linear inseparable dataset

Table 4. Experimental results of four algorithms on linear inseparable dataset

Algorithm	Evaluation Indicators (mean±standard deviation)					
	Accuracy	AUC	F_mean	G_mean	Recall	Precision
CSEL	97.27±1.81	96.87±1.58	97.52±1.83	92.86±2.44	97.16±0.31	98.43±0.24
BalanceCascade	96.73±2.72	92.68±5.84	96.22±2.76	88.86±4.85	96.89±1.33	97.51±0.31
EasyEnsemble	95.55±2.06	90.52±4.39	95.54±2.81	87.63±3.96	96.27±1.27	97.28±1.56
Adaboost	89.64	83.43	93.97	84.77	96.98	88.99

In order to verify the difference between formula (19) and formula (20), ten experiments were carried out with formula (19) and formula (20) respectively. Table 5 is the mean and standard deviation of ten experiments with formula (19) and formula (20). We can see that the results of formula (19) are better than formula (20).

Table 5. Experimental results with formula (19) and formula (20)

formula	Evaluation Indicators (mean ±standard deviation)					
	Accuracy	AUC	F_mean	G_mean	Recall	Precision
Formula (19)	97.27±1.81	96.87±1.58	97.52±1.83	92.86±2.44	97.16±0.31	98.43±0.24
Formula (20)	96.85±2.01	96.04±1.77	96.93±2.18	92.41±2.36	96.98±0.59	97.89±0.56

4.2. Experimental Results on UCI Dataset

This section carries out simulation experiments on 7 UCI datasets (Car, Breast cancer, Haberman's survival, Blood transfusion, Contraceptive, Teaching and Tic-Tac-To). The experimental results of the proposed algorithm and three others algorithm (BalanceCascade, EasyEnsemble and AdaBoost) show in table 6. Every UCI dataset is divided into test dataset and training dataset. The samples in two datasets are randomly selected from the corresponding UCI dataset. Table 6 show the detailed sample distribution of the experimental dataset. We can see that there is a high imbalance degree in the training datasets and the test datasets are all close to the balance datasets. In addition, the algorithm proposed in this paper mainly aim at binary classification, so all experimental datasets are transformed into dataset only include two classes.

Table 6. Experiment results on UCI data set

Dataset	Attribute	Training dataset			Test dataset		
		Majority	Minority	Proportion	Majority	Minority	Proportion
Car	9	400	100	4.0	226	232	1
breast	9	100	25	4	101	60	1.7
haberman	3	150	30	5	75	51	1.5
blood	4	450	50	9	120	128	1
Contraceptive	9	644	100	6.44	200	200	1
Teaching	5	70	20	3.5	32	28	1
Tic-Tac-To	9	400	100	4	226	228	1

Table 7 shows the mean and standard deviation of ten experiments results on the above seven UCI datasets using CSEL algorithm, BalanCecascade algorithm, EasyEnsemble and AdaBoost algorithm. Table 5 shows the detailed experimental results about evaluate dictators of *Accuracy*, *AUC*, *F-means*, *G-means*, *Recall* and *Precision*. Fig.5, Fig.6, Fig.7 and Fig.8 show the ROC curve of CSEL algorithm, BalanceCascade algorithm, EasyEnsemble algorithm and AdaBoost algorithm on above seven datasets.

Table 7. Experimental result of four algorithms on UCI dataset

Dataset	Alogrithm	Evaluation Indicators (mean±standard deviation)					
		Accuracy	AUC	F-mean	G-mean	Recall	Precision
teaching	CSEL	**70.28±1.03**	**71.55±0.54**	**73.62±0.87**	**68.44±1.35**	**65.74±0.74**	**72.15±0.91**
	BalanceCascade	64.05±2.64	67.84±0.78	48.48±1.92	56.57±3.15	49.02±3.47	64.18±2.18
	EasyEnsemble	68.43±3.00	64.52±0.34	68.30±1.01	60.47±2.80	63.18±0.68	65.37±1.85
	AbaBoost	68.39	61.75	63.26	64.51	63.75	66.81
car	CSEL	**82.24±1.73**	82.11±1.21	**77.42±0.98**	**80.07±1.19**	**75.04±1.38**	**81.73±2.15**
	BalanceCascade	81.70±2.71	**82.87±2.84**	75.08±3.17	76.46±2.86	70.24±2.09	75.71±3.61
	EasyEnsemble	72.23±1.94	77.28±1.91	65.54±2.54	70.42±2.05	64.47±1.84	62.87±3.49
	AbaBoost	70.83	75.33	60.50	66.62	70.87	49.32
breast	CSEL	**68.53±1.57**	**72.37±1.53**	**60.55±1.72**	**68.34±1.97**	78.95±1.74	**72.79±2.39**
	BalanceCascade	65.73±2.84	70.18±2.06	56.38±3.35	67.82±3.01	**48.08±2.85**	75.04±4.21
	EasyEnsemble	62.36±2.44	68.50±1.83	51.36±3.24	63.63±2.90	81.99±2.88	68.09±6.91
	AbaBoost	60.23	65.41	35.29	49.61	74.36	48.57
haberman	CSEL	**73.86±1.08**	**90.80±2.31**	**68.25±0.81**	**79.77±1.75**	83.64±1.56	**78.01±1.08**
	BalanceCascade	74.05±1.74	87.03±1.66	64.44±0.97	71.86±1.04	82.67±1.39	77.89±1.67
	EasyEnsemble	67.78±1.96	76.15±0.96	53.90±1.34	70.20±1.28	**84.77±0.98**	75.79±3.07
	AbaBoost	65.82	67.70	27.45	41.79	77.86	28.42
blood	CSEL	**70.18±0.61**	**82.56±0.83**	50.84±1.04	**71.08±0.98**	82.58±1.15	**70.35±2.04**
	BalanceCascade	67.37±2.34	78.21±1.52	45.85±1.97	63.98±2.16	**83.37±1.96**	65.17±3.51
	EasyEnsemble	66.20±1.80	72.20±0.77	49.44±1.49	67.22±1.38	83.26±1.05	69.44±3.47
	AbaBoost	69.95	63.18	**50.98**	63.23	80.87	43.82
Contraceptive	CSEL	**69.24±0.76**	**75.85±0.81**	**60.33±0.95**	**70.19±0.83**	**81.72±0.57**	**75.41±1.66**
	BalanceCascade	66.45±1.26	71.63±1.32	51.32±1.83	67.38±1.64	80.05±1.21	73.03±2.76
	EasyEnsemble	66.77±0.84	74.23±0.93	53.57±1.21	68.59±1.05	81.02±0.95	72.87±2.35
	AbaBoost	68.32	66.39	46.55	57.98	80.41	36.00
Tic-Tac-Toe	CSEL	**80.58±0.97**	**79.86±0.83**	**66.67±0.92**	**72.42±0.84**	82.30±0.89	**63.52±1.18**
	BalanceCascade	78.08±2.18	78.56±1.51	65.32±1.75	69.32±1.62	80.42±1.73	56.02±2.37
	EasyEnsemble	78.81±1.33	76.87±1.09	65.28±1.48	70.85±0.99	80.89±1.42	60.96±1.29
	AbaBoost	79.75	48.38	63.53	69.56	**84.50**	50.90

From table 7, in most of the experimental results of most indicators (*Accuracy*, *AUC*, *F-means*, *G-means*, *Recall* and *Precision)*, the experimental results of proposed algorithm are relatively better than that of AdaBoost algorithm, EasyEnsemble algorithm and BalanceCascade algorithm except on the indicator of Recall. The average classification accuracy of the propose algorithm increases by 2.17%. The average AUC of the proposed algorithm increases by 2.84%. The average *F-mean* and *G-mean* of the

propose algorithm increases by 7.18% and 5.27% respectively. The classification accuracy of minority class is greatly improved. The average *Recall* and *Precision* of the proposed algorithm increase by 2.87% and 3.84% respectively.

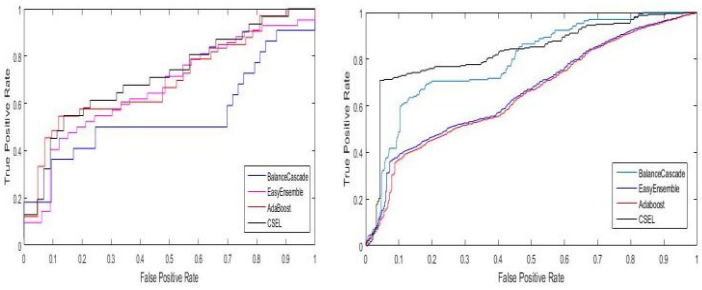


Fig.8. ROC curve of Teaching and Car dataset

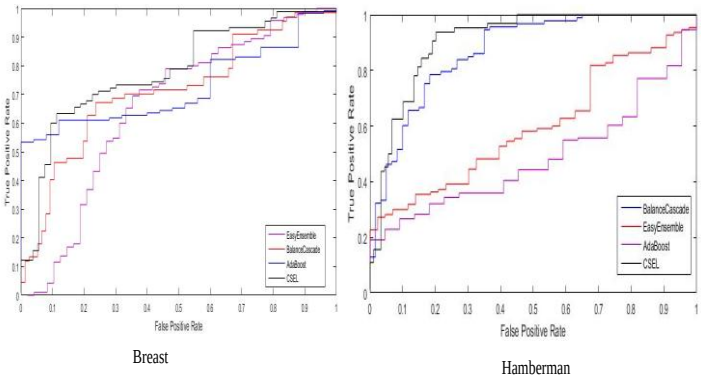


Fig.9. ROC curve of Breast and Hamberman dataset

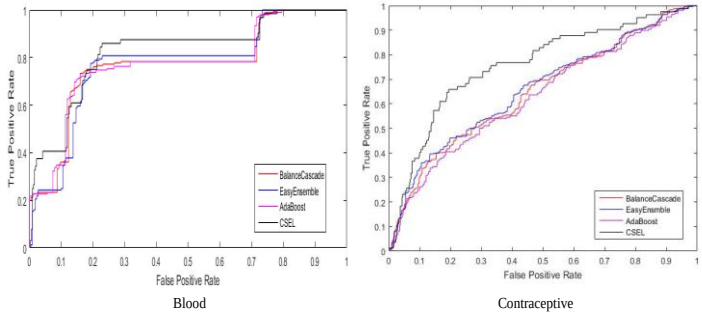


Fig.10. ROC curve of Blood and Contraceptive dataset

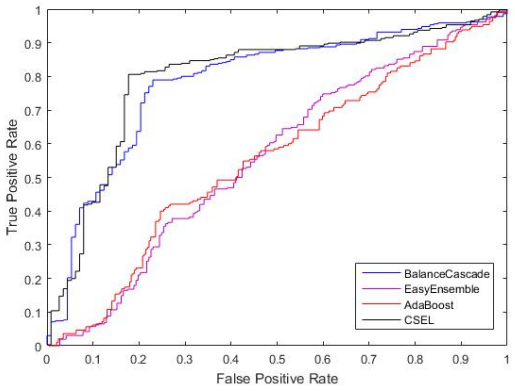


Fig.11. ROC curve of Tic-Tac-Toc dataset

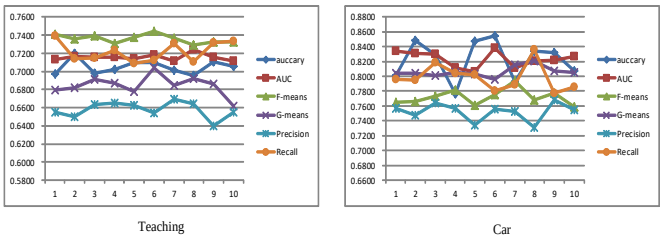


Fig.12. Change curve of ten experiments of Teaching and Car dataset

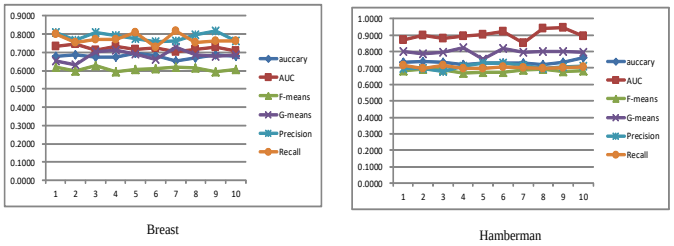


Fig.13. Change curve of ten experiments of Breast and Hamberman dataset

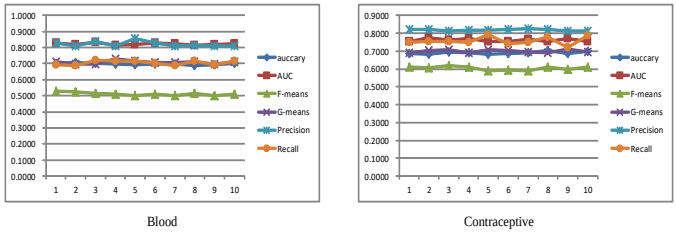
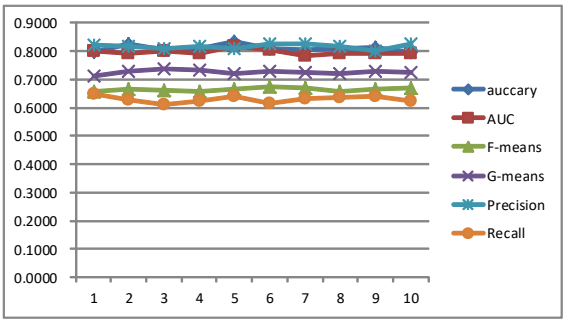


Fig.14. Change curve of ten experiments of Blood and Contraceptive dataset



Tic-Tac-Toc

Fig.15. Change curve of ten experiments of Tic-Tac-Toc dataset

Fig. 8-11 show the ROC curve of seven UCI datasets. Compared with EasyEnsemble algorithm, BalanceCascade algorithm and AdaBoost algorithm, CSEL algorithm has better classification performance on evaluate indicator AUC. From the experimental data in Table 7 and seven ROC curve, CSEL algorithm has better experimental results for imbalanced dataset. And it has a high precision of minority class. Because we pay more attention to the recognition rate and false alarm rate of minority class samples in practical application, so it meets the needs of practical application.

The change curve of each evaluate indicator of ten experiments on sever UCI datasets is shown in Fig.12, Fig.13, Fig.14 and Fig.15. From Fig.12, it can be seen from the figure that every evaluate indicator vary greatly on Teaching and Car dataset. However, in other UCI datasets, the change of all indicators is relatively gentle, which shows that the algorithm is relatively stable from Fig.13, Fig.14 and Fig.15.

4.3. Experimental results on KDDCUP dataset

KDDCUP99 dataset is an imbalanced dataset, because there are a large number of normal behaviour data and a small amount of attack behavior data in practical application. It includes training dataset and test dataset. Training set includes 494 022 samples and test set includes 31000 samples. There are 24 kinds of attack records in training dataset and 38 kinds of attack records in test dataset. The algorithm proposed in this paper is oriented to class binary-classification problem. Therefore, all attack samples are looked as abnormal class and all normal samples are looked as normal class. The experimental data in this paper were selected at equal intervals to obtain 5176 samples as test data and 2674 samples as training data to maintain the original dataset space distribution. The detailed distribution of experimental data is shown in Table 6. From the training dataset, we can see that the experimental dataset is imbalanced.

This section still compares the performance with AdaBoost algorithm, BalanCecascad algorithm and EasyEnsemble algorithm on KDDCUP dataset [29]. As can be seen from table 6, test dataset includes some attack type samples that do not include in the training data. The purpose is to verify the detection of new attacks. The experimental performance has been compared with the above seven indicators.Table 8 shows the detailed data of the experiment about evaluate dictators of *Accuracy*, *AUC*, *F-*

means, *G-means*, *Recall* and *Precision*. The ROC curve of a random experiment is shown in Fig. 9 of four algorithms (CSEL algorithm, BalanceCascade algorithm, EasyEnsemble algorithm and AdaBoost algorithm).

From table 9, we can see that the test data is an imbalanced dataset and the training data is an imbalanced dataset that the imbalanced ratio between the number of majority class and it of minority class is 10.3. The attack behaviour samples are looked as minority class and the normal behaviour samples are looked as majority class in the training data set. Test dataset includes 12 kinds of attack samples and 24 kinds of unknown attack samples. In addition, the test data is an approximately balanced dataset. Therefore, it will have a high classification error rate of minority class if the classification Hyperplane offset to the minority class. The proposed selective ensemble learning algorithm for imbalanced data is suitable for anomaly detection of network intrusion. It is important for anomaly detection of network intrusion to reduce the classification error rate of minority class, meanwhile, to reduce the missing alarm rate and false alarm rate.

Table 8. Experiment dataset of KDDCUP

Data type	dataset	
	Test dataset	Train dataset
Total samples	5176	2674
Normal samples	2146	2438
Attack samples	2582	236
Unknown samples	448	0
Attack types	28	17

Table 9. classification accuracy of four algorithms on KDDCUP dataset

algorithm	Detailed accuracy of every type of data (mean± standard deviation)				
	All records	Normal record	Intrusion record	Known intrusion	Unknown intrusion
CSEL	85.58±0.31	91.21±0.37	81.78±0.31	83.04±0. 23	74.55±0.45
BalanceCascade	82.08±0.97	91.99±0.44	75.05±0.48	75.91±0.37	70.09±0.89
EasyEnsemble	79.14±0. 8	91.38±0.21	70.43±0.32	72.11±0.34	62.72±0.24
Adaboost	80.12	90.86	72.48	73.74	65.18

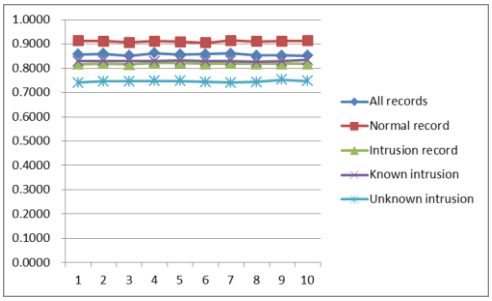


Fig.16. Accuracy change curve of ten experiments of KDDCUP dataset

Table 9 is classification accuracy of all kinds of data (include overall accuracy, normal type, intrusion type, unknown intrusion type and known intrusion type). It can be seen that classification accuracy of intrusion samples is greatly increased from table

9, especially classification accuracy of unknown intrusion samples. The detection rate of unknown intrusion is increased by 4.46% compare with BalanceCascade algorithm. Classification accuracy of known intrusion is increased by 7.13%. It can be seen that the proposed algorithm may increase the classification accuracy of the minority class and total samples from table 9. In other words, it may reduce false alarm rate and missed alarm rate. Fig.16 shows the change curve of classification accuracy of various data (includes: All records, Normal records, Intrusion records, Known intrusion, Unknown records, intrusion records) in ten experiments. From the change curve, we can see that in ten experiments, the indicators are relatively stable, indicating that the algorithm is relatively stable.

Table 10. Experimental results of KDDCUP dataset

Algorithm	Evaluation Indicators (mean ±standard deviation)					
	Accuracy	AUC	F_mean	G_mean	Recall	Precision
CSEL	85.58±0.31	87.52±0.26	80.57±0.18	83.54±0.23	81.97±0.12	68.94±0.29
BalanceCascade	82.08±0.85	84.26±1.16	80.15±0.95	82.69±0.74	80.76±0.97	65.82±1.24
EasyEnsemble	79.14±0.18	83.39±0.35	78.72±0.24	82.18±0.17	81.92±0.23	63.96±0.43
Adaboost	80.12	84.18	79.19	83.04	83.76	63.89

The detailed experimental results about accuracy rate, *F-mean*, *G-mean*, *AUC*, *Recall* and *Precision* is shown in table 10. It can be seen that the experimental results of the most evaluate indicators are better compared with BalanceCascade algorithm, EasyEnsemble algorithm and Adaboost algorithm from table 10, especially the *recall* and *precision* of intrusion behavior (minority class). This is due to the proposed algorithm pay more attention on sample distribution.

The ROC curves about four algorithms are random one experiment on KDDCUP dataset is shown in Fig. 18. The change curve of each evaluate indicator of ten experiments on KDDCUP dataset is shown in Fig.18. We can see that the change of all indicators is relatively gentle from Fig.18, which shows that the algorithm is relatively stable.

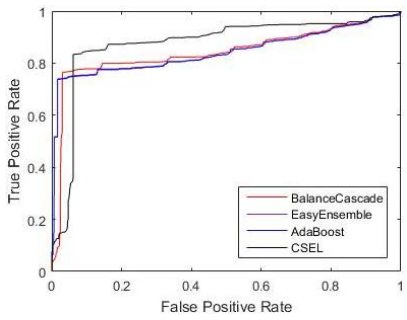


Fig.17. ROC curve of KDDCUP dataset

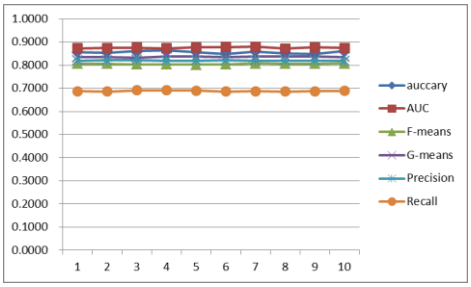


Fig.18. Change curve of ten experiments of KDDCUP dataset

5. Conclusion and Discussion

Ensemble learning algorithm can also be used to solve the problem of imbalanced data. It mostly uses some methods to re-sample the majority samples and obtains a dataset equivalent to the number of samples in a minority class. Then, a training subset is formed with the minority class samples. A certain classification algorithm is used to train the base-classifier. Repeat many times to get multiple base-classifiers, and then ensemble multiple base-classifiers. A selective ensemble learning algorithm for imbalanced data is proposed to reduce the influence of data imbalance in this paper. The simulation results on artificial dataset, UCI dataset and KDDCUP dataset show that the proposed algorithm can reduce the effect of imbalanced data.

Acknowledgment. This work is supported by Key Research and Development Program of Shaanxi of China (Program No.2022GY-073); Science and Technology Plan Project of Shangluo City of China (No. 2021-J-0002); Science and Technology Innovation Team Building Project of Shangluo University of China (No.18SCX002).

References

1. Du H.L., Zhang Y., Ke G., et al. (2021) A selective ensemble learning algorithm for imbalanced dataset, Journal of Ambient Intelligence and Humanized Computing, DOI.10.1007/s12652-021-03453-w.
2. Wan J.W., Yang M., Chen Y.J (2012) Const Sensitive Semi-Supervised Laplacian Support Vector machine [J]. ACTA ELECTRONICA SINICA of China, 40(7):1410-1415
3. Duan L.X., Guo H.,Wang J.J (2016) A mechanical fault severity identification method under unbalanced datasets [J]. JOURNAL OF VIBRATION AND SHOCK of China, 35(20):178-182
4. Liu L., Wang B., Zhong Q., et al. (2015) A selective ensemble method based on K-means method[C]// International Conference on Computer Science & Network Technology. IEEE, 2015.
5. Zhou Y.H., Zhou Z.H (2016) Large margin distribution learning with cost interval and unlabeled data [J]. IEEE Transactions on Knowledge and Data Engineering,2016,28(7):1749-1763

6. Li Y.J., Guo H.X., Li Y.N., etc. (2016) A Boosting based Ensemble Learning Algorithm in Imbalanced Data Classification[J]. Systems Engineering-Theory & Practice of China,2016,36(1):189-199
7. Xing S.,Wang X.H., Wang X.L (2016) Extreme Learning Machine Ensemble Learning based on Multi Class Resampling for imbalanced Data[J].JOURNAL OF NANJING UNIVERSITY (NATURAL SCIENCES) of China,2016,52(1):203-211
8. Jian, C., Gao, J., Ao, Y (2016) A new sampling method for classifying imbalanced data based on support vector machine ensemble [J]. Neurocomputing, 2016,193(1), 115–122
9. Wang Q., Luo Z.H., Huang J.C, et al. (2017) A Novel Ensemble Method for Imbalanced Data Learning: Bagging of Extrapolation-SMOTE SVM [J]. Computational intelligence and neuroscience, 2017, 2017(3):1827016.
10. Wu S., Liu L., Lu D (2017) Imbalanced Data Ensemble Classification based on Cluster-based Under-sampling Algorithm[J]. Chinese Journal of Engineering, 2017, 39(08):1244-1253.
11. Wang S., Minku L.L., Yao X (2015) Resampling-based ensemble methods for online class imbalance learning[J]. IEEE Transactions on Knowledge and Data Engineering,2015,27(5):1356-1368
12. Zhang, H., Li, J. L,, Liu, X. M., et al. (2021) Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection[J]. Future Generation Computer Systems, 122: 130-143.
13. Chen, N. N., Gong, X. T., Wang, Y. M., et al. (2021) Random clustering forest for extended belief rule-based system [J]. Soft Computing, 2021, 25(6): 4609-4619.
14. Liu X.Y., Wu J., Zhou Z.H (2009) Exploratory undersampling for class-imbalance learning [J]. IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics, 2009, 39(2): 539-550.
15. Guo H., Li Y., Li Y., et al. (2016) BPSO-Adaboost-KNN ensemble learning algorithm for multi-class imbalanced data classification[J]. Engineering Applications of Artificial Intelligence, 2016, 49(C):176-193
16. Shipp C.A., Kuncheva L.I (2002) Relationships between combination methods and measures of diversity in combining classifiers [J]. Information Fusion,2002,3(2):135-148
17. Brown G (2009) An information theoretic perspective on multiple classifier systems[C]. //Proc of the 8th International Workshop on Multiple Classifier Systems,2009:344-353
18. Zhou Y.H., Zhou Z.H (2016) Large margin distribution learning with cost interval and unlabelled data [J]. IEEE Transactions on Knowledge and Data Engineering,2016,28(7):1749-1763
19. Zhang C.X., Zhang J.S (2011) A survey of selective ensemble learning algorithms [J]. Jisuanji Xuebao/chinese Journal of Computers, 2011, 34(8):1399-1410.
20. Zhang Y., Liu B., Yu J (2017) A selective ensemble learning approach based on evolutionary algorithm[J]. Journal of Intelligent & Fuzzy Systems, 2017, 32(3):2365-2373.
21. Potharaju S.P., Sreedevi M (2017) Ensembled Rule Based Classification Algorithms for predicting Imbalanced Kidney Disease Data [J]. Journal of Engineering Science & Technology Review, 2017, 9(5):201-207.
22. Zhai J., Zhang S., Wang C (2017) The classification of imbalanced large data sets based on MapReduce and ensemble of ELM classifiers [J]. International Journal of Machine Learning & Cybernetics, 2017, 8(3):1009-1017.
23. Haque M.N., Noman N., Berretta R., et al. (2016) Heterogeneous Ensemble Combination Search Using Genetic Algorithm for Class Imbalanced Data Classification[J]. PLOS one, 2016,11(1): e0146116
24. Wan J.W., Yang M., Chen Y.J (2012) Const Sensitive Semi-Supervised Laplacian Support Vector machine [J]. ACTA ELECTRONICA SINICA of China, 40(7):1410-1415
25. Zhong S, Chen T, He F, et al. (2014) Fast Gaussian kernel learning for classification tasks based on specially structured global optimization [J]. Neural Networks, 2014, 57: 51-62.

26. Zhang Y., Du H.L (2019) Imbalanced Heterogeneous Data Ensemble Classification based on HVDM-KNN [J]. CAAI Transactions on Intelligent Systems of China, 2019,14(4):733-742
27. KUNCHEVA L.I., WHITAKER C. J (2003) Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy[J]. Machine Learning,2003,51(2):181-207
28. Yu H. and Ni J (2014) An improved ensemble learning method for classifying high-dimensional and imbalanced biomedicine data [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics,2014,11(4):1339-1347
29. Du H.L., Zhang Y., Ke G., et al. (2021) Online ensemble learning algorithm for imbalanced data stream, Applied Soft Computing,Volume 107,2021, https://doi.org/10.1016/j.asoc.2021.107378.

Hongle Du received the B.S. degree in computer science and technology from Henan University, Kaifeng, China, in 2004 and the M.S. degree in computer application technology from Guangdong University of Technology, Guangzhou, China, in 2010. His current research interests include machine learning, pattern recognition and network security.

Yan Zhang received the B.S. degree in computer science and technology from Xidian University, Xi'an, China, in 2004 and the M.S. degree in computer application technology from Northwest A&F University, Xi'an, China, in 2011. His current research interests include machine learning, pattern recognition and network security.

Lin Zhang is a professor in School of Mathematics and Computer Application, Shangluo University. He received the B.S. degree in computer science and technology from Shaanxi Normal University, Xi'an, China, in 1991. His current research interests include machine learning, Online learning and network security.

Yeh-Cheng Chen is a PhD at the Department of Computer Science, University of California, Davis, CA, USA. His research interests are radio-frequency identification (RFID), data mining, social network, information systems, wireless network artificial intelligence, IoT and security.

Received: August 17, 2022; Accepted: January 15, 2022.

Computer Science and Information Systems 20(2):857–878 https://doi.org/10.2298/CSIS220706024Y

Personalization Exercise Recommendation Framework based on Knowledge Concept Graph

Zhang Yan[1,2,3], Du Hongle[1,2,4], Zhang Lin[1,2], and Zhao Jianhua[1,2,3]

[1] School of Mathematics and Computer Application, Shangluo University, 726000 Shangluo, China
{dhl5597, flyingzhang83, zhlin002}@163.com
[2] Shangluo Public Big Data Research Center, 726000, Shangluo, China
[3] Engineering Research Center of Qinling Health Welfare Big Data, Universities of Shaanxi Province,726000 Shangluo, China
[4] College of Information Technology and Computer Science, University of the Cordilleras, 2600 Baguio City, Philippines

Abstract. With the explosive increase of online learning resources, how to provide students with personalized learning resources and achieve the goal of precise teaching has become a research hotspot in the field of computer-assisted teaching. In personalized learning resource recommendation, exercise recommendation is the most commonly used and most representative research direction, which has attracted the attention of a large number of scholars. Aiming at this, a personalized exercise recommendation framework is proposed in this paper. First, it automatically constructs the relationship matrix between questions and concepts based on students' answering records (abbreviated as Q-matrix). Then based on the Q-matrix and answer records, deep knowledge tracing is used to automatically build the course knowledge graph. Then, based on each student's answer records, Q-matrix and the course knowledge graph, a recommendation algorithm is designed to obtain the knowledge structure diagram of every student. Combined the knowledge structure diagram and constructivist learning theory, get candidate recommended exercises from the exercise bank. Finally, based on their diversity, difficulty, novelty and other characteristics, exercises are filtered and obtain the exercises recommended to students. In the experimental part, the proposed framework is compared with other algorithms on the real data set. The experimental results of the proposed algorithm are close to the current mainstream algorithms without the Q-matrix and curriculum knowledge graph, and the experimental results of some indicators are better than Algorithms exist.

Keywords: Personalization Exercise Recommendation, Course Knowledge graph, Deep Knowledge Tracing, knowledge concept.

1. Introduction

With the continuous launch of Massive Online Open Course (MOOC), in the face of massive online learning resources, how to choose suitable learning resources is an urgent problem to be solved in the current intelligent education system [1]. The intelligent education system uses computer-aided technology to help students obtain personalized and suitable learning resources from the massive online teaching

resources, thereby improving the teaching effect. It can dynamically adjust the learning resources recommended to each student according to the student's learning trajectory (learning style, knowledge mastery, etc.). In the intelligent education system, the course knowledge graph plays an important role in the recommendation of learning resources, learning path recommendation, etc., and the knowledge graph of the curriculum is also a hot spot in this field [2].

The course knowledge graph (Some scholars call it a knowledge graph, and some scholars call it a concept map, which is called a knowledge graph in the subsequent chapters of this article.) represents the dependencies between the knowledge concepts of a course. Combined with constructivist learning theory, students should learn a course according to the relationship between knowledge concepts in the knowledge graph. This is because the learning sequence of knowledge concepts affects students' learning effect [3]. If the course knowledge graph is accurately obtained, it can intuitively show the student's mastery of the knowledge concepts in one course, which can promote students to carry out meaningful learning, and can also draw students' own knowledge according to each student's mastery of knowledge concepts. In addition, it also can be used to recommend the personalized learning resources.

In the existing personalized learning resource recommendation algorithms, knowledge graphs are mostly marked by experts, on the one hand, the workload is large, on the other hand, there will be a certain deviation. In response to this, many experts and scholars have carried out corresponding research on the course knowledge graph, and proposed corresponding construction methods of the course knowledge graph [4-5]. The references [6-10] mainly use association rule mining algorithm and traditional statistical theory to build the course knowledge graph, and the references [11-13] use the machine learning method to build the course knowledge graph. Whether it is an association rule based on statistical theory or a method based on deep learning, it is assumed that the exercise-knowledge concept relationship matrix (*Q-matrix*) already exists. In practical applications, the acquisition of *Q-matrix* also requires expert annotation, and there are also problems of heavy workload and labeling deviation.

In view of the above problems, a personalized exercise recommendation framework based on course knowledge graph is proposed in this paper. First, based on the students' answer records, the exercise-knowledge concept relationship matrix is automatically obtained by using deep knowledge tracing and improved K-means clustering algorithm. Using students' answer records and *Q-matrix*, design a knowledge graph construction algorithm, and automatically obtain the course knowledge graph. Then, based on students' answer records and *Q-matrix*, a knowledge graph construction algorithm is designed, and automatically obtain the course knowledge graph. Then combine the *Q-matrix*, the knowledge graph and the students' answer records to draw the knowledge structure map of each student (that is, the mastery degree map of each knowledge concept). Then, based on the constructivist learning theory, the exercise recommendation model is constructed according to the course knowledge graph, the student knowledge structure map, and the *Q-matrix*. Then, the recommended exercises are selected according to the characteristics of the difficulty, novelty, and diversity of the exercises.

This study provides a framework for personalized learning resource recommendation based on course knowledge graph, and discusses how to automatically construct the exercise-knowledge concept relationship matrix and the relationship map between knowledge concepts (course knowledge graph). On this basis, construct the knowledge

structure diagram of each student, and design a recommendation model to recommend personalized learning resources for each student (this article mainly focuses on exercises). The main contributions include the following three-fold: (1) An idea of automatically constructing the course knowledge graph is proposed. The relationship matrix between exercises and knowledge concepts is automatically constructed, and then the course knowledge graph is automatically constructed. (2) This paper proposes a personalized exercise recommendation framework based on the curriculum knowledge graph, and presents a new idea of recommending personalized learning resources based on the knowledge graph, combining the acquired curriculum knowledge graph, student answer records and *Q-matrix*. Then, the knowledge structure diagram of each student is obtained according to the answer record of each student. Then, combine the two diagrams and the *Q-matrix*, build a recommendation model, and recommend personalized exercises for every student. (3) In this paper, the calculation method of the importance of knowledge concepts is given for each student, and then selects exercises related to the most important knowledge concepts for students from many exercises to recommend.

The remaining chapters of this paper is organized as follows: The second subsection mainly analyzes the research progress of the existing personalized recommendation algorithm, and analyzes the work that still needs to be improved and the improved method. The third subsection mainly introduces the recommendation framework proposed in this paper, and introduces the problems solved in each stage and the methods to solve them in detail. The fourth subsection mainly introduces the personalized exercise recommendation model based on knowledge graph and the available recommendation methods. The fifth subsection presents several common algorithms and experimental results on common data sets, and compares and analyzes the proposed framework and the corresponding experimental results. The sixth subsection discusses and analyzes the conclusions and the main work of the next stage.

2. Related Works

This section reviews the research on personalized exercise recommendation algorithms, knowledge graph construction, and knowledge tracing. The basic ideas and implementation processes of various methods will be introduced, then analyzes the advantages and disadvantages of various methods, and proposes corresponding solutions.

2.1. Knowledge Tracing

Knowledge tracing is based on the modeling of students' behavior sequences to predict students' mastery of knowledge concept. Commonly used knowledge tracing models include Bayesian Knowledge Tracing (BKT) [15] and Deep Knowledge Tracing (DKT) [16]. In most knowledge tracing and improvement methods, the sequence of answering questions is used to train the model, however, the sequential relationship between knowledge concepts is ignored.

According to the knowledge transfer theory in the field of education [17], when a learner learns a knowledge concept, it not only changes the proficiency of the existing knowledge concept, but also changes the mastery of the associated knowledge concept. Therefore, the curriculum knowledge graph that shows the dependencies between knowledge concepts has attracted the attention of many experts and scholars [18-21]. Many scholars build the corresponding model for the knowledge graph and study the problem of knowledge tracing. Inspired by the success of graph neural network (GNN), Nakagawa et al. [18] applied GNN to knowledge tracing task for the first time, and proposed a Graph-based Knowledge Tracing (GKT). This method converts the knowledge structure into a graph, thereby indirectly reconstructing the knowledge tracing task into a time series node-level classification problem in the GNN model. Graph convolutional neural networks (GCNS) are proposed for semi-supervised graph classification, updating self-node representations based on information about itself and its neighbors.

Therefore, if multiple graph convolution layers are used, the updated node representation contains the attributes of neighbor nodes and the information of high-order neighbors. Yang et al. [19] proposed a knowledge graph tracing (GIKT) along this line of thought. To make full use of the textual information contained in the exercises and the hierarchical relationship between the exercise and knowledge concepts. Tong et al. [20] proposed a hierarchical graph knowledge tracing model framework (HGKT). The method defines a problem schema to calculate the similarity between exercises. Then, a hierarchical graph neural network is proposed based on the problem mechanism. It adopts two attention mechanisms to emphasize the important historical states of students. The hierarchical graph and sequences are full used to improve the performance of knowledge tracing. Zhu et al. [21] builds a knowledge graph based on knowledge units, target knowledge units, knowledge unit dependencies, etc. Then, multiple learning paths can be obtained, and then judges the learning progress of learners according to their learning logs, and learning paths are recommended to them. Shi et al. [22] constructed a learning goal-oriented knowledge graph across learning domains, including six semantic relations, and then combined the learner's learning goals and the characteristics of learning resources to represent the recommended learning path.

Most of the existing methods assume that the knowledge graph of the course already exists, and then build a knowledge tracing model based on the knowledge graph, answer records, and *Q-matrix* and other information. However, how to automatically and accurately obtain the knowledge graph and *Q-matrix* corresponding to the course is the key to this type of method.

2.2. Course Knowledge Graph

The curriculum knowledge graph reflects the relationship graph between the curriculum knowledge concepts, and represents the dependency between the knowledge concepts in the curriculum. According to the knowledge transfer theory [17] and the constructivism theory [32], the curriculum knowledge graph is very important in the intelligent education system. Yu et al. [4] use association rule mining algorithm to analyze the relationship between curriculum knowledge concepts, and automatically builds a knowledge graph between curriculum knowledge concepts. The key to this method lies

in the setting of association rules and how to set the thresholds of each indicator, so that the pruned knowledge graph can more accurately represent the relationship between course knowledge concepts. In this article, in order to obtain the knowledge graph, the thresholds of various indicators are directly specified.

In practical applications, according to the granularity of knowledge concept division, there may be hundreds of knowledge concepts, and the key to construct the knowledge graph is each threshold. The author later improved this method [5-6]. In this method, the exercise-knowledge concept relationship matrix (*Q-matrix*) needs to be manually labeled, which not only requires a large workload, but also makes the labeling subjective and has labeling bias. In response to this problem, the literatures [7-9] use the combination of association rules and machine learning methods to construct the knowledge graph of the course. Shao et al. [7] combine text recognition and association rule mining to automatically obtain exercises related to each knowledge concept and automatically construct a *Q-matrix*. It can avoid the bias caused by manual labeling of the *Q-matrix*. This method uses the association rule mining algorithm in the literature [6] to build a knowledge graph, but it also has the problem that it is difficult to determine multiple thresholds. Huang et al. [8] uses soft rules and adversarial learning ideas to automatically build a knowledge graph, which can reduce the impact of thresholds on algorithm performance and improve the quality of the composition. Literatures [11-13] combine the ideas of deep learning and data mining to automatically build a knowledge graph to avoid the impact of setting thresholds on the algorithm. Lin et al. [11] mainly conduct research on knowledge tracing. Assuming that there is only one question related to a knowledge concept, this method can obtain the relationship map between knowledge concepts.

However, in practical applications, one knowledge concept corresponds to multiple exercises, and one exercise corresponds to multiple knowledge concepts. In response to this, Zhang et al. [12] uses a hierarchical attention mechanism and a relational graph neural network to build a knowledge graph (Relational Graph neural network with Hierarchical Attention). Hiromi et al. [13] presents six methods for constructing curriculum knowledge graph. Among them, three methods based on multi-head attention mechanism, based on Variational autoencoder (VAE) and based on probability transition matrix have better performance, and can automatically obtain the knowledge graph of the corresponding course in a given answer record and *Q-matrix*. Obviously, this method still needs Manually annotated *Q-matrix*.

2.3. Exercise Recommendation Algorithm

Building an exercise recommendation model [22-23] is crucial in an intelligent education system. Traditional methods based on product recommendation mainly include content-based recommendation algorithm [22-23] and collaborative filtering recommendation algorithm [24-25]. In recent years, graph neural networks have attracted the attention of many experts and scholars. Using graph neural networks to solve recommendation problems is a hot spot in current recommendation system research [26-28]. In order to reflect the dependencies between knowledge concepts, Lin et al [26] proposed a knowledge graph smart extraction and explicit diagnosis (CM-SEED) learning system. Use knowledge graph to describe students' mastery of knowledge. In order to more accurately describe the mastery of knowledge concept and

the dependencies between knowledge concepts, Lv et al. [27] proposed an intelligent exercise problem recommendation method based on weighted knowledge graph. This method quantifies students' learning ability with students' test data, and then uses the dependence between students' learning ability and knowledge concepts to improve the effect of personalized exercise recommendation. The knowledge graph proposed in papers [26-27] is actually the cognitive structure graph of each student, not the knowledge graph of the course.

In order to make full use of the dependencies between the nodes of the knowledge graph, literatures [28-30] modeled on the course knowledge graph to realize personalized learning resource recommendation. Huang et al. [28] combines knowledge graph and collaborative filtering, and the use of collaborative filtering based on knowledge graph can not only effectively avoid the problems of data sparseness and cold start of traditional collaborative filtering recommendation methods, but also avoid the inconsistency of K value in traditional collaborative filtering recommendation algorithms. Wang et al. [29] use Graph Convolutional Networks for modeling based on knowledge graph to mine the associations between nodes on the knowledge graph, automatically discover structural information and semantic information. Then, the neighbor information of each node is used to describe the node, that is, the interrelated path between nodes can be obtained. Finally, Knowledge Graph Convolutional Networks is used to implement the recommendation of personalized learning resources. The current GNN-based models are mostly coarse-grained in relational modeling. They do not identify user-item relationships at the fine-grained graph level, and do not use relational dependencies to preserve the semantics of remote connections.

Knowledge Graph-based Intent Network (KGIN) [30] strengthens the independence of different user-item relationships, thereby enhancing the ability and interpretability of the model. A new information aggregation mechanism is used to recursively integrate relational sequences (relationship paths) of remote connections. Zhao et al. [31] use Recurrent Neural Networks (RNNs) to predict the coverage of knowledge concepts based on students' answer records, and uses Deep Knowledge Tracing (DKT) to predict students' mastery of knowledge concepts. Then, according to the two prediction results, a set of exercises recommended to students is obtained, and then a certain strategy is used to filter the set of exercises to obtain the exercises that are finally recommended to students. This method ignores the transfer of dependencies between knowledge concepts, and also does not consider the influence of the difficulty of the exercises.

3. Personalized Exercise Recommendation Framework Based on Knowledge Graph

Based on the above analysis, this section will introduce the proposed framework of personalized exercise recommendation in detail. The framework starts from the students' answering records, uses a certain method to automatically construct the *Q-matrix* (such as using DKT), and then automatically constructs the course knowledge graph course based on the *Q-matrix*. Then, combining each student's answer record, *Q-matrix* and course knowledge graph, each student's knowledge structure map can be got, and finally build a recommendation model based on the knowledge structure map and get the personalized exercise.

3.1. Recommended Framework

The personalized exercise recommendation framework based on the curriculum knowledge graph proposed in this paper is shown in Fig. 1. It is mainly divided into three stages: (1) construct a curriculum knowledge graph based on students' learning behavior data; (2) Obtain students' mastery of personal knowledge, construct the knowledge structure graph of students; (3) build a personalized exercise recommendation model, and get personalized exercise recommendation for each student. The following is a detailed introduction to the personalized exercise recommendation framework proposed in this paper from above three stages.

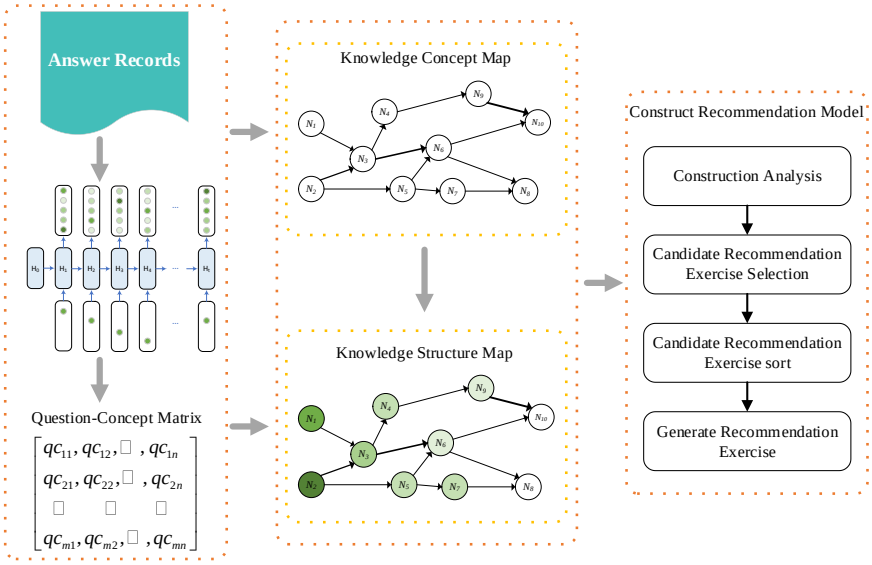


Fig. 1. Personalized Exercise Recommendation Framework

3.2. Automatically Obtain the *Q-matrix*

The curriculum knowledge graph can accurately reflect the dependencies between curriculum knowledge concepts. According to the knowledge transfer theory [17] and the constructivism theory [32], when the learner's learning and mastery of the current concept will be affected by the related knowledge concept, it will also affect the mastery of the related knowledge concept. In other words, the learning sequence of knowledge concepts has an impact on the learning effect. So, what order is a reasonable order? This is the problem to be solved by the curriculum knowledge graph. The literatures [19, 33] believe that the knowledge graph is a hierarchical tree structure. For example, each textbook is divided into several chapters, each chapter includes several sections, and so on, so there is a hierarchical tree structure between the course knowledge concepts. Some other scholars also believe that the curriculum knowledge graph is a network structure [28-30], and there is a sequential dependency between knowledge concepts. It

is a directed acyclic network structure graph, that is, there are predecessor knowledge concepts and successor knowledge concepts. Learning process should be based on the order of knowledge concepts, which is more in line with the knowledge transfer theory and constructivism theory in pedagogy, so that students can obtain good learning effects.

The question now is how to obtain an accurate course knowledge graph. The course knowledge graph in the literatures [19, 28, 30, 33] is obtained by expert annotation. Due to the personal subjectivity of expert annotation, there will be deviations in the annotated knowledge graph. In response to this problem, the literatures [4-8] use association rule mining, deep learning, natural language analysis and other methods to automatically build a curriculum knowledge graph. But these methods are all constructing the course knowledge graph given the *Q-matrix*, and the construction of the *Q-matrix* is also obtained by the expert annotation. According to the different granularity of knowledge concepts, generally each course includes hundreds of knowledge concepts, and each course includes thousands of exercises (especially for middle school courses), and each exercise involves one or more knowledge concepts. Therefore, constructing the *Q-matrix* requires a lot of work, and there are biases.

Based on the above analysis, this paper presents a method to automatically construct the *Q-matrix*, The deep knowledge tracing model constructed in [15] can obtain the sequential dependencies between exercises, and the exercises are related to knowledge concepts. Assuming that each exercise is only related to one knowledge concept, the exercise related to a knowledge concept can be classified into one category through cluster analysis, thus obtaining the *Q-matrix*.

Assuming that m students have n exercises, the answer records of m students can be expressed as a matrix:

$$A=\begin{bmatrix} a_{11}, a_{12}, \mathrm{L}\ , a_{1n} \\ a_{21}, a_{22}, \mathrm{L}\ , a_{2n} \\ \mathrm{M} \quad \mathrm{O} \quad \mathrm{M} \\ a_{m1}, a_{m2}, \mathrm{L}\ , a_{mn} \end{bmatrix}$$

Where $a_{ij}=1$ denotes that the *i*-th student answers the *j*-th exercise correctly, and $a_{ij}=0$ denotes that the *i*-th student answers the *j*-th exercise incorrectly.

According to the students' answer records, the dependence of *i*-th exercise on *j*-th exercise can be expressed as the formula (1):

$$A_{i,j}=\frac{y(j\mid i)}{\sum_k y(j\mid k)} \tag{1}$$

Where $y(j\mid i)$ denotes the probability that *j*-th exercise is correct when *i*-th exercise is answered correctly.

Assuming that students have mastered the knowledge concepts, then the students can correctly do the exercises related to the knowledge concepts, otherwise they cannot correctly do the exercises related to the knowledge concepts. Therefore, after doing *i*-th exercise correctly and then doing *j*-th exercise correctly, then *i*-th exercise and *j*-th exercise either belong to the same knowledge concept, or the knowledge concept corresponding to *i*-th exercise is the precursor of the knowledge concept corresponding to *j*-th exercise. In the same way, after answering *i*-th exercise incorrectly, answering *j*-

th exercise incorrectly is the same. Based on this, when *K*-means clustering is performed, the distance calculation method can be expressed as the formula (2):

$$dis(vecA, vecB) = \sum_{i=1}^{n} (vecA_i \text{ e } vecB_i) \quad (2)$$

Assuming that there are *k* knowledge concepts in total, after *k*-means clustering, the question-concept matrix can be obtained as follows:

$$Q = \begin{bmatrix} qc_{11}, qc_{12}, \text{L} , qc_{1n} \\ qc_{21}, qc_{22}, \text{L} , qc_{2n} \\ \text{M} \quad \text{O} \quad \text{M} \\ qc_{k1}, qc_{k2}, \text{L} , qc_{kn} \end{bmatrix}$$

Where $qc_{ij} = 1$ denotes that the *j*-th exercise is related to the *i*-th knowledge concept, and $qc_{ij} = 0$ denotes that the *j*-th exercise is not related to the *i*-th knowledge concept.

3.3. Building a Knowledge Graph

After the *Q-matrix* is generated, the course knowledge graph can be constructed based on the *Q-matrix* and the students' answer records. Literatures [4-6] use the association rule mining algorithm to automatically build the knowledge graph. This method first defines the association rules, and then calculates the confidence of the rules according to the formula (3).

$$conf(Q_i \to Q_j) = \frac{\sup(Q_i, Q_j)}{\sup(Q_i)} \quad (3)$$

Then specify the minimum confidence (the value in this article is 0.75), the difference degree threshold the difference degree is 0.35, and several other thresholds. By setting a good threshold, the graph is trimmed to obtain the final knowledge graph. In this method, the setting of the threshold is very important to the constructed knowledge graph. The setting of the threshold in this paper is specified based on experience and cannot be applied to a dynamically changing environment. In response to this, the literatures [7-9] use machine learning methods to build knowledge graphs. Nakagawa et al. [18] provides two methods to calculate the relationship between two nodes, one is a statistical-based method. This method builds the relationship between neighbor nodes based on the adjacency matrix *A*, and $f_{neighbor}$ adopts formula (4) to calculate:

$$f_{neighbor}(h_i^t, h_j^t) = A_{i,j} f_{out}([h_i^t, h_j^t]) + A_{j,i} f_{in}([h_j^t, h_i^t]) \quad (4)$$

where f_{out} and f_{in} are multilayer perceptron（MLPs）, the key is the construction method of $A_{i,j}$. For examples:

(1) $$A_{i,j} = \begin{cases} \dfrac{n_{i,j}}{\sum_k n_{i,k}} & i \neq j \text{（Transition probability matrix）} \\ 0 & i = j \end{cases} \quad (5)$$

(2) $$A_{i,j} = \frac{y(j \mid i)}{\sum_k y(j \mid k)} \text{ (deep knowledge tracing)} \quad (6)$$

$y(j \mid i)$ denotes the probability that the *j*-th exercise is correct when the *i*-th exercise is answered correctly (using the RNN network).

(1) Multi-head attention (MHA): The multi-head attention mechanism is used to calculate the weight of the edge between two nodes. $f_{neighbor}$ can calculate with the formula (7).

$$f_{neighbor}(h_i^t, h_j^t) = \frac{1}{K} \sum_{k \in K} a_{ij}^k f_k(h_i^t, h_j^t) \tag{7}$$

a_{ij}^k is the *k*-th head's attention weight from v_i to v_j, and f_k is a neural network for the *k*-th head.

(2) Variational autoencoder (VAE). According to the type of edge represented by the node feature, it is calculated by the following formula (8):

$$f_{neighbor}(h_i^t, h_j^t) = \sum_{k \in K} z_{ij}^k f_k(h_i^t, h_j^t) \tag{8}$$

Where, z_{ij}^k is a latent variable sampled from the Gumbel–Softmax distribution. f_k is a neural network for the k-th edge type.

In the experimental stage, three methods of association rule, DKT and MHA are used to verify the validity of the proposed exercise recommendation framework.

3.4. Student's Personal Knowledge Structure Diagram

Analyzing the degree of knowledge mastery of students is a crucial step in the intelligent education system. Only by accurately identifying the degree of knowledge mastery of each student can we accurately recommend personalized exercises. Methods such as Bayesian Knowledge Tracing [13] and Deep Knowledge Tracing [14, 15] are commonly used to predict students' mastery of knowledge concept. Since knowledge graphs can reflect the dependencies between knowledge concepts, literatures [18-20] use graph neural networks to analyze students' mastery of knowledge concept based on knowledge graphs. The interpretability of these existing methods is poor, and it is not easy to understand the students' mastery of each knowledge concept. Therefore, this paper analyzes each student's mastery of the curriculum knowledge concepts based on the curriculum knowledge graph, and first defines several description knowledge concepts.

Definition 1. The mastery of knowledge concepts *MC*, the correct rate of students who have done exercises related to knowledge concepts is used to indicate the mastery of knowledge concepts for each student, using formula (9) to calculate:

$$\mathrm{MC_i} = \frac{\sum_{QC_{a_{j,i}}=1, a_j=1} QC_{a_{j,i}}}{\sum_{QC_{a_{j,i}}=1} QC_{a_{j,i}}} \tag{9}$$

Where, $Q_{a_{j,i}} = 1$ denotes the exercise a_j is related to i-th knowledge concept. $Q_{a_{j,i}} = 1, \mathrm{a_j} = 1$ denotes that the exercise a_j is related to the *i*-th knowledge concepts are answered correctly.

Combining knowledge transfer theory [17] and constructivism theory [32], the order in which students learn knowledge concepts has an impact on the students' mastery of knowledge concepts, and what has a direct impact on the current knowledge concept is the predecessor knowledge concept. That is, the mastery of the precursor knowledge concept will affect the students' learning effect of the current knowledge concept.

Definition 2. Knowledge concept support degree *SP*, the number of the predecessor knowledge concepts that have been mastered divided by the number of all predecessor knowledge concepts, assuming that the precursor knowledge concept set of the *i*-th knowledge concept is expressed as *SC*, the support degree of the current knowledge concept can be expressed for formula (10):

$$SP(N_i) = \frac{\sum_{N_j \in SC} MC_j}{|SC|} \tag{9}$$

Where, $SP(N_i) = 1$ denotes that the predecessor knowledge concept of this knowledge concept has been fully mastered and should be recommended first.

When recommending exercises to students, it actually selects a knowledge concept from the course knowledge graph, and then recommends exercises related to the knowledge concept. That is to say, knowledge concepts with higher importance should be selected for recommendation. How to evaluate the importance of knowledge concepts? Combined with the characteristics of the directed graph, the importance of each node (knowledge concept) is related to the in-degree, out-degree and the weight of the corresponding node and edge. For individual students, the higher the mastery of knowledge concepts, the lower the importance of knowledge concepts; the greater the out-degree of a knowledge concept, the greater the importance of the knowledge concept as the precursor of multiple knowledge concepts; if the greater the in-degree of a knowledge concept, it means that the more restricted knowledge concepts, the recommendation should be postponed, that is to say, the lower the mastery of the knowledge concept. Based on the above analysis, the definition of the importance of the knowledge concept is given as follow.

Definition 3. The importance degree of knowledge concept *CP*. The importance of the knowledge concept is inversely proportional to the mastery of the knowledge concept, and inversely proportional to the difference between the node in-degree and the node's support degree, which can be calculated by formula (11).

$$CP(N_i) = \begin{cases} (1-MC_i)\dfrac{out\_\deg ree(N_i)}{in\_\deg ree(N_i)(1-SP(N_i))}, in\_\deg ree(N_i) \neq 0 \\ (1-MC_i) * out\_\deg ree(N_i) \quad , in\_\deg ree(N_i) = 0 \end{cases} \tag{11}$$

Where $out\_\deg ree(N_i)$ and $in\_\deg ree(N_i)$ denote the out-degree and in-degree of node N_i. The importance of knowledge concepts is actually the priority of recommended knowledge concepts, and is the main basis for selecting recommended knowledge concepts.

The importance of knowledge concepts is for students, and each knowledge concept is of different importance to different students. In the algorithm of this paper, the knowledge concepts to be recommended and the corresponding exercises are selected according to the importance of the knowledge concepts. It is mainly considered from the following three aspects: (1) the importance of the knowledge concepts that students

have mastered is less than that of the knowledge concepts that they have not mastered, because the knowledge concepts that are not mastered will affect the learning of subsequent knowledge concepts; (2) the importance of knowledge concept that has a large out-degree is greater than that of a knowledge concept with a small out-degree, because a large out-degree indicates that its mastery will affect the learning of multiple knowledge concepts, so the importance is large; (3) In-degree and support degree also have an impact on the importance of a knowledge concept. The greater the in-degree, the more affected factors. The greater the support degree *SP* of a knowledge concept, the better the mastery of the precursor knowledge concept of the knowledge concept.

3.5. Recommendation algorithm based on knowledge graph

After obtaining the course knowledge graph, build a recommendation model based on the knowledge graph. Designing recommendation models based on graph neural networks is a hot topic in current recommendation system research [26-28]. Due to the poor interpretability of deep learning, it is difficult to display the recommendation results intuitively. In this regard, this paper based on the course knowledge graph and each student Based on the Knowledge Structure Map (KSM), combined with the answer records and *Q-matrix*, a simple personalized exercise recommendation algorithm is designed. The algorithm flow is shown in Fig. 2.

The detailed process of the algorithm is described as follows:

Algorithm 1. Exercise recommendation algorithm based on knowledge graph

```
Input: Course Knowledge Graph CKG, Student Answer Record
A, Question-Concept Relationship Matrix Q
Output: Recommended exercises
For i=1:K
Calculate the MCi
If MCi<threshold
    Calculate CP(Ni)
Else
    Continue
Sort CP(Ni) in descending order
Choose the recommended exercise based on A matrix and Q-
matrix
```

First, according to the knowledge graph and the students' personal answer records, the students' mastery degree MC_i of each knowledge concept is obtained. If the knowledge concept has been mastered, it will not be processed. Otherwise, the importance degree $CP(N_i)$ corresponding to the knowledge concept is calculated. Then sort $CP(N_i)$ in descending order, select the first knowledge concept N_j, and then according to the *Q-matrix*, adopt a certain strategy to select the exercises related to the knowledge concept N_j and recommend them to the students. Here are two kinds of choose strategy.

(1) Random selection, randomly select one or several exercises from the exercises related to the knowledge concept N_j and recommend them to students;

(2) According to the degree of mastery of the knowledge concept, the recommended exercises are selected according to the accuracy of all students' answers to the exercises related to the knowledge concept and the students' individual mastery of the knowledge

concept. Assuming that there are K exercises related to the knowledge concept N_i, first calculate the answering accuracy rate P_k of the relevant exercises, and then calculate the mastery of the knowledge concept CP_i, and select the exercises with the smallest difference between (1-P_k) and CP_i to be recommended. If the knowledge concept is poorly mastered, the exercises with low difficulty are recommended, and if the knowledge concept is well mastered, the more difficult exercises are recommended.

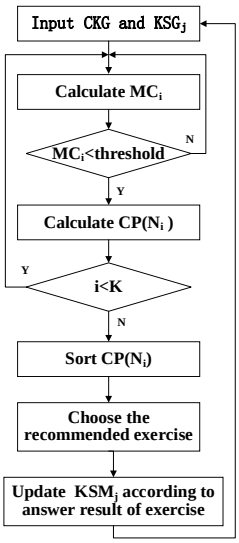


Fig. 2. The algorithm flow char

4. Experiments and Analysis

This subsection mainly verifies the effectiveness of the proposed method through experiments, and shows the comparison of various experimental data and experimental results under different data sets. The following will introduce the experimental process in detail and analyze the experimental results in detail.

4.1. Experimental data sets

The method proposed in this paper constructs the *Q-matrix* based on the assumption that an exercise is only related to one knowledge concept. In order to fully verify the framework, in this section, all the exercises in the dataset are regarded as related to only one knowledge concept. The data sets used in the experiments are described in detail below.

- *synthetic* dataset. The dataset [33] is machine simulation data, and the response performance of each student is generated using the problem difficulty, student ability and probability of random guessing based on IRT. Contains 4,000 students and 50 exercises, each with 5 knowledge

points, for a total of 200,000 practice records.

- *ASSISTments 2009* data set. This dataset [33] is collected from the ASSISTments online education platform in the 2009-2010 school year, and contains 325,673 practice records of 4,151 students on 110 knowledge points after deduplication. The duplicate records of the source dataset are removed before the experiment. At the same time, the dataset has the problem of data sparsity, and its density is only 0.06.
- *ASSISTment 2015* data set. This dataset contains 708,631 practice records of 19,840 students on 100 knowledge points. Due to the large learner base, the dataset contains a small number of student records on average. This dataset is the sparsest among the available datasets, with a density of only 0.05.
- *Intellilence 2018* dataset. This dataset provided by Oneoftops Ltd., a company that provides teaching and learning services to teachers and students in the Advanced Center, includes 256,612 learning records from 1,452 students across 159 knowledge points.

Table 1. Experimental dataset

Dataset	KCs	Students	Records
Synthetic	25	4000	200000
ASSISTments 2009	110	4151	325673
ASSISTments 2015	74	19840	525535

Table 1 gives the overall description of the data set used in the experiment. The algorithm in this paper assumes that each exercise is only related to one knowledge concept. In the Synthetic data set, each subset has only 50 exercises. In order to increase the number, 5 datasets are merged, that is, 250 exercises, including 25 knowledge concepts. In the ASSISTments data set, the order of answering questions for each student is different, and the order of answering questions has an impact on the learning effect of each student, but in order to use the clustering algorithm to obtain the *Q-matrix*, the exercises done by each student are Sorting, in the answer record, the correct value of the exercises that each student has not done is 0.

4.2. Base Line

The prediction accuracy (ACC) and the area under the ROC curve (AUC) are two evaluation indicators that are often used to evaluate the classification performance. In this experiment, these two evaluation indicators are used to evaluate the effectiveness of the proposed framework. This paper compares the experimental results of the traditional classical algorithms BKT and DKT. In addition, the algorithm framework proposed in this paper is based on the knowledge graph, and the DKVMN and GKT algorithms are the classical methods in the graph neural network. Therefore, this paper mainly compares these four algorithms. KCG_MHA and KCG_VAE are the frameworks proposed in this article. The MAHA algorithm and VAE algorithm are used to build the knowledge graph. KCG_TPM and KCG_DKT use the Transportation Probability Matrix and Deep Knowledge Tracing to build the knowledge graph. The experimental

results of the algorithm proposed in this paper are all without a given *Q*-matrix and knowledge graph.

BKT [15]: BKT is the traditional method based on probability. It predicts the switches between every two knowledge concepts according to the HMM rules and knowledge master degree.

DKT [16]: DKT is the first using an RNN to deal with the problem of knowledge tracing. It predicts students' next attempts on knowledge concept based on the students' knowledge mastery degree.

GKT [18]: This method adopts the knowledge graph to predict the students' knowledge mastery degree. Knowledge graphs can reflect the relationships among knowledge concepts. Therefore, this method predicts recommendation according to the GNN model.

HGKT [20]: HGKT is the improved method of GKT, and it introduces the hierarchical idea and multi-layer attention mechanism into the model to make prediction.

KCG_MHA: This adopts the above-mentioned the recommendation framework. But the adjacency matrix is constructed based on the muti-head attention mechanism. Then knowledge graph is constructed based adjacency matrix.

KCG_VAE: This adopts the above-mentioned the recommendation framework. But the adjacency matrix is constructed based on the Variational autoencoder. Then knowledge graph is constructed based adjacency matrix.

KCG_TPM: This adopts the above-mentioned the recommendation framework. But the *Q-matrix* is constructed based on the transportation probability matrix.

KCG_DKT: This adopts the above-mentioned the recommendation framework. But knowledge graph is constructed based on the deep knowledge tracing.

4.3. Experimental Results and Analysis

In our experiment, 60% of the data is used for training, and the remaining 40% of the data is used for testing. Due to the contingency in the division of the data set, the experiment in this paper uses ten random divisions of the data set. The data are the average of ten experiments results, and Figures 3, 4 and 5 are the average of ten experiments on three datasets.

Fig.3 shows the results of Accuracy and AUC on Synthetic dataset. Fig.4 shows the results of Accuracy and AUC on ASSISTments 2009 dataset. Fig.5 shows the results of Accuracy and AUC on ASSISTments 2015 dataset. The three figures intuitively show the experimental performance of 8 methods (including the four methods proposed in this paper, KCG_MHA, KCG_VAE, KCG_TPM and KCG_DKT) on three datasets. From the three figures, you can see the performance of the method proposed in this paper. The performance is better than the BKT method, and the performance is slightly worse than that of the HGKT method. This is because the method in this paper only uses the answer records in the data set, while the other four methods (BKT, DKT, GKT and HGKT) apply the answer records at the same time, and *Q*-matrix, and the acquisition of *Q*-matrix requires expert annotation.

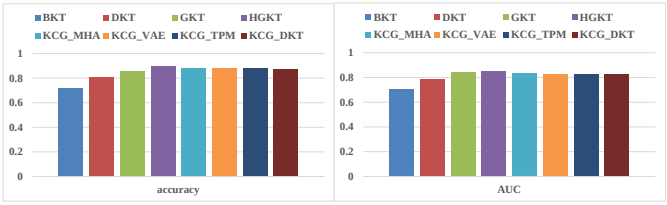


Fig.3. Experimental comparisons of Synthetic dataset

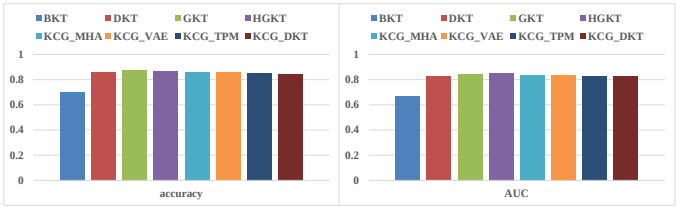


Fig.4. Experimental comparisons of ASSISTments 2009 dataset

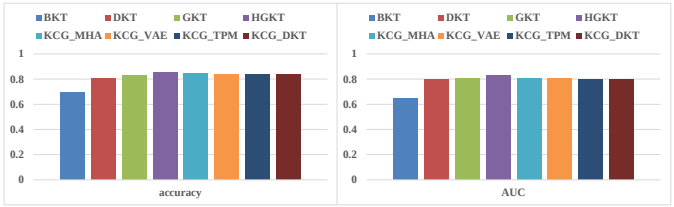


Fig.5. Experimental comparisons of ASSISTments 2015 dataset

Table 2. Experimental results of Accuracy and AUC

Model	*Synthetic*		*ASSISTments 2009*		*ASSISTments 2015*	
	ACC	AUC	ACC	AUC	ACC	AUC
BKT	0.7201	0.7036	0.7014	0.6714	0.6977	0.6511
DKT	0.8125	0.7876	0.8577	0.8263	0.8101	0.7953
GKT	0.8575	0.8443	0.8741	0.8461	0.8341	0.8102
HGKT	0.8991	0.8564	0.8715	0.8497	0.8515	0.8295
KCG_MHA	0.8812	0.8385	0.8633	0.8381	0.8433	0.8091
KCG_VAE	0.8796	0.8276	0.8574	0.8344	0.8407	0.8078
KCG_TPM	0.8788	0.8243	0.8501	0.8278	0.8376	0.7993
KCG_DKT	0.8703	0.8294	0.8449	0.8266	0.8395	0.8021

The four methods given in this paper build the course knowledge graph. The KCG_TPM method is used for the course knowledge graph on the Synthetic dataset and the ASSISTments 2015 dataset as shown in Fig. 6 and Fig. 8. After obtaining the course knowledge graph, according to the formula (11) can obtain the knowledge structure diagram of each student in the corresponding course, and you can clearly see the students' mastery of each knowledge concept. Fig. 7 shows the knowledge structure diagram of random two students' mastery of each knowledge concept on the Synthetic dataset. Fig. 9 shows the knowledge structure diagram of the random two students'

mastery of each knowledge concept on the ASSISTments 2015 data set. The color depth in the figure represents the mastery of the knowledge concept, and the darker the color, the mastery of the knowledge concept. The better, the lighter the color, the worse the mastery.

From Fig. 3, we can see the dependencies between the various knowledge concepts in the course. Based on the course knowledge map, a certain path selection algorithm can be used to recommend learning paths and recommended exercises for students. Similarly, using the dependencies between courses can recommend courses for students. This paper mainly provides personalized exercise recommendation for students based on the curriculum knowledge map. Based on the knowledge graph shown in Fig. 3, Algorithm 1 is used to recommend exercises. The recommended effect is shown in Table 1. As can be seen from Table 1, on the Synthetic dataset, the performance of the four methods for building knowledge graphs provided in this paper is relatively close. The algorithm is better than the BKT, DKT and GKT methods, but the performance is 2.88%-1.79% different from the HGKT algorithm. On the one hand, in the HGKT algorithm, the *Q-matrix* is already given and accurate, which makes the constructed course knowledge graph more accurate. On the other hand, using a hierarchical graph neural network to implement recommendation on the knowledge graph requires more computing resources, but if given appropriate parameters, it will have better experimental performance.

Figures 7 and 9 are the knowledge structure diagrams of any two students on the two datasets respectively. Through the students' personal knowledge structure diagram, we can clearly see the mastery of each knowledge concept of each student. For example, Fig. 7(a) and Fig. 7(b) represent the mastery degree of knowledge concept of student a and student b respectively. Comparing student a and student b, it can be seen that student b has a better mastery of course knowledge concepts than student a (The color of multiple nodes in Figure b is darker than that of the nodes in Figure a, that is, the mastery of the corresponding knowledge concepts is better). Specifically, comparing the two figures, it can be seen that student b's mastery of knowledge concept 8 and knowledge concept 12 is better than student a, and student b's mastery of knowledge concepts 25 and 14 is slightly better than student a.

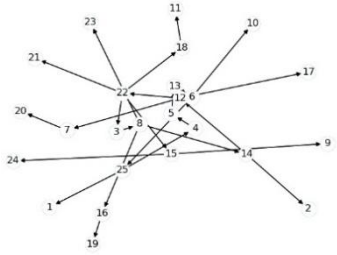
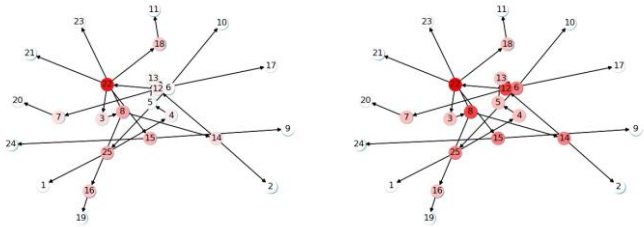


Fig. 6. The knowledge graph of Synthetic dataset



a b

Fig.7. The knowledge structure diagrams of any two students on Synthetic dataset

As can be seen from Fig. 6, there is a loop on the knowledge graph, that is to say, there is an interdependent relationship between knowledge concepts. For example, there is a ring between knowledge concepts 4, 5, and 25, that is to say, in order to master knowledge concept 4 well, you need to master knowledge concept 25; to master knowledge concept 25 well, you need to master knowledge concept 5, and mastering knowledge concept 5 requires mastering knowledge concept 4. In practical applications, the course knowledge graph should be a directed acyclic graph, so this situation should not exist. In order to eliminate the loops in the knowledge graph, we try to use the spanning tree protocol method in the computer network to prune each loop in the knowledge graph. Using the pruned knowledge graph for exercise recommendation, the recommendation performance is relatively poor. This is because this paper assumes that each exercise is only related to one knowledge concept, but in practice, one exercise is related to multiple knowledge concepts. The method proposed in this paper is used to construct *Q-matrix* and curriculum knowledge graphs, a loop will appear.

The curriculum knowledge graph of the ASSISTments 2015 dataset is shown in Fig. 8. The total number of knowledge concepts are 74, which is relatively large. Therefore, the use of circles can better display the dependencies between knowledge concepts. From the figure, we can also see that there is partial loop phenomenon. Fig. 9 shows the knowledge structure diagram of two random students, in which the shade of color indicates the degree of mastery of the knowledge concept.

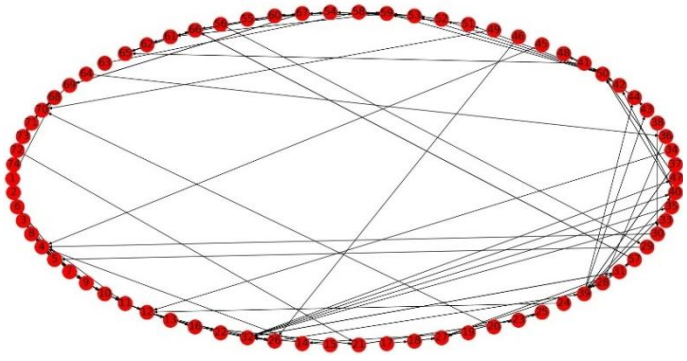
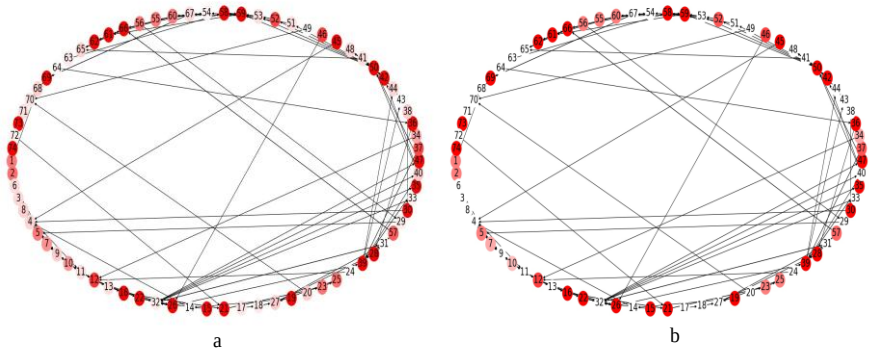


Fig.8. The knowledge graph of ASSISTments 2015 dataset



a b

Fig.9. The knowledge structure diagrams of any two students on ASSISTments 2015 dataset

5. Conclusion

This paper attempts to build a personalized exercise recommendation framework based on knowledge graphs, which does not require manual adoption. This method starts from constructing a question-concept matrix(Q-matrix), then builds a curriculum knowledge graph based on the *Q-matrix* and student answer records, and then builds student's personal knowledge structure graph based on the course knowledge graph, *Q-matrix* and students answering records, and then uses each student's knowledge structure graph to recommend personalized exercises for students. The recommendation framework makes the recommendation process intuitive, and you can intuitively see the mastery of each knowledge concept and the recommended exercises for each student. In addition, the framework is also suitable for recommending personalized learning paths for students. The experimental results show that this method has better performance than BKT, and has better intuitiveness than DKT and GKT. However, the framework proposed in this paper also has certain limitations. (1) The method in this paper assumes that each exercise is only related to one knowledge concept, but in practical applications, each exercise is related to multiple knowledge concepts, then how to implement the framework Many-to-many relationship between exercises and knowledge concepts. The process of constructing the *Q-matrix* is actually the process of classifying each exercise (category is the concept of knowledge). Aiming at this problem, the method of semi-supervised classification is proposed to assign exercises to multiple classes, and determine which class should belong to according to the probability of belonging to each class. (2) When recommending exercises, the recommended exercises are determined according to the importance of each knowledge concept to the students. This method is relatively intuitive, but its performance is not good. For this, a graph neural network is proposed to implement personalized exercise recommendation. How to solve the problem? The above two questions will be the main work in the next stage.

Acknowledgment. This work is supported by Key Research and Development Program of Shaanxi of China (Program No.2022GY-073); Shaanxi Provincial Education Department scientific research program Foundation of China (No.21JK0616); Science and Technology Plan Project of Shangluo City of China (No. 2021-J-0002); Science and Technology Innovation Team Building Project of Shangluo University of China (No.18SCX002).

References

1. Li Q., Zhong S., Wang P., Guo X., Quan X.: Learner model in adaptive learning system, J. Inf. Comput. Sci., vol. 7, no. 5, pp. 1137–1145, (2010)
2. li Y.C., Shao Z.Z., Wang X., Zhao X.C., Guo Y.H.: A Knowledge Graph-Based Learning Paths Automatic Generation Algorithm for Adaptive Learning Systems, 2018 IEEE Access. (2018)
3. Grundspenkis J.: Development of knowledge graph based adaptive knowledge assessment system, in Proc. IADIS Multi Conf. Comput. Sci. Inf. Syst., e-Learn. (MCCSIS), 2008, pp. 395–402. (2008)
4. Yu, Z., Zheng, X., Huang, F., et al.: A framework based on sparse representation model for time series prediction in smart city[J]. Frontiers of Computer Science, 2021,15(1): 1-13. (2021)
5. Chen S.M., Bai S.M.: Using data mining techniques to automatically construct knowledge graphs for adaptive learning systems. Expert Systems with Applications. 2010, doi: 10.1016/j.eswa.2009.12.060. (2010)
6. Wang, S., Wang, Z., Lim, K. L., Xiao, G. and Guo, W.: Seeded random walk for multi-view semi-supervised classification [J]. Knowledge-Based Systems, 2021, 222: 107016. (2021)
7. Shao Z.Z., Li Y.C., Wang X., Zhao X.C., Guo X.C.: Research on a new automatic generation algorithm of knowledge graph based on text analysis and association rules mining. Journal of Ambient Intelligence and Humanized Computing, 2020,11:539–551. https://doi.org/10.1007/s12652-018-0934-9. (2020)
8. Huang X., Yang K., Lawrence V.: Classification-based approach to knowledge graph generation in adaptive learning, in Proc. IEEE 15th Int. Conf. Adv. Learn. Technol., Jul. 2015, pp. 19–23, doi: 10.1109/ICALT.2015.149. (2015)
9. Tang C.F., Rao Y., Yu H.L, Sun L., Cheng J.M., Wang Y.T.: Improving Knowledge Graph Completion Using Soft Rules and Adversarial Learning. Chinese Journal of Electronics. 2021, DOI:10.1049/cje.2021.05.004. (2021)
10. Acharya A., Sinha D.: An intelligent Web-based system for diagnosing student learning problems using knowledge graphs, J. Educ. Comput. Res.,vol. 55, no. 3, pp. 323–345, 2016, Doi: 10.1177/0735633116667358. (2016)
11. Lin J.J., Zhao Y.Z., Huang W.Y., Liu C.F., Pu H.T.: Domain knowledge graph-based research progress of knowledge representation. Neural Computing and Applications,2021, 33:681–690. https://doi.org/10.1007/s00521-020-05057-5. (2021)
12. Zhang, Z., Zhuang, F., Zhu, H., Shi, Z., He, Q.: Relational graph neural network with hierarchical attention for knowledge graph completion. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(5), 9612-9619. (2020)
13. Hiromi N., Yusuke I., and Yutaka M.: Graph-based Knowledge Tracing: Modeling Student Proficiency Using Graph Neural Network. In IEEE/WIC/ACM International Conference on Web Intelligence (WI'19), October 14–17, 2019, Thessaloniki, Greece. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3350546.3352513. (2019)
14. Yudelson M.V., Koedinger K.R., Gordon G.J.: Individualized Bayesian Knowledge Tracing Models[C]//16th International Conference on Artificial Intelligence in Education (AIED 2013). Springer, Berlin, Heidelberg. (2013)
15. Piech C., Spencer J., Huang J.: Deep Knowledge Tracing[J]. Computer Science, 2015, 3(3): pags. 19-23. (2015)
16. Du H.L., Zhang Y., Ke G., Zhang L., Chen Y.C.: Online ensemble learning algorithm for imbalanced data stream, Applied Soft Computing, Volume 107,2021, https://doi.org/10.1016/j.asoc.2021.107378. (2021)
17. Liang K., Ren Y.M., Shang Y.H., Zhang Y.Y., Wang C.: Review of Knowledge Tracing Preprocessing Based on Deep Learning. 2017, Computer Engineering and Applications (in China). (2017)

18. Nakagawa H., Iwasawa Y., Matsuo Y.: Graph-based Knowledge Tracing: Modeling Student Proficiency Using Graph Neural Network[C]//Proc of the Int Conf on Web Intelligence (WI). Piscataway, NJ: IEEE, 2019: 156-163. (2019)
19. Yang Y., Shen J., Qu Y.: GIKT: a graph-based interaction model for knowledge tracing[J]. arXiv preprint arXiv:2009.05991, 2020:68-79. (2009)
20. Tong H., Zhou Y., Wang Z.: HGKT: introducing problem schema with hierarchical exercise graph for knowledge tracing[J]. arXiv e-prints, 2020: arXiv: 2006.16915. (2006)
21. Zhu H., Tian F., Wu K.: A multi-constraint learning path recommendation algorithm based on knowledge graph[J]. Knowledge-Based Systems, 2018,143(12):102-114. (2018)
22. Shi D., Wang T., Xing H.: A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning[J]. Knowledge-Based Systems, 2020,195(2):105618. (2020)
23. Seng, D.W., Chen, X L., Fang, X J., Zhang, X F., Chen, J.: Research on Personalized Recommendation of Educational Resources Based on Big Data. Educational Sciences: Theory & Practice, 2018, 18(5), 1948-1959. (2018)
24. Li X.Y., Chen C.H., Zheng P.: A context-aware diversity-oriented knowledge recommendation approach for smart engineering solution design. Knowledge-Based Systems. 2021, https://doi.org/10.1016/j.knosys.2021.106739. (2021)
25. Li G., Yi G., Zhen Y.: A user-knowledge dynamic pattern matching process and optimization strategy based on the expert knowledge recommendation system. Applied Intelligence.2021, https://doi.org/10.1007/s10489-021-02289-3. (2021)
26. Lin Y.H., Chang Y.C., Liew K.H.: Effects of knowledge graph extraction and a test-based diagnostic environment on learning achievement and learners' perceptions, British Journal of Educational Technology.2016, doi:10.1111/bjet.12250. (2016)
27. Lv P., Wang X.X., Xu J.: Intelligent personalized exercise recommendation: A weighted knowledge graph-based approach. 2020, DOI: 10.1002/cae.22395. (2020)
28. Huang G., Yuan M., Li C.S., Wei Y.H.: Personalized Knowledge Recommendation Based on Knowledge Graph in Petroleum Exploration and Development. International Journal of Pattern Recognition and Artificial Intelligence. 2020, DOI: 10.1142/S0218001420590338. (2020)
29. Wang H.W., Zhao M., Xie X., Li W.J, Guo M.Y.: Knowledge Graph Convolutional Networks for Recommender Systems. In Proceedings of the 2019 World Wide Web Conference (WWW '19), 2019, May 13–17, 2019, San Francisco, CA, USA. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3308558.3313417. (2019)
30. He X.N., Deng K., Wang X., Li Y., Zhang Y.D., Wang M.: LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. SIGIR '20, July 25–30, 2020, Virtual Event, China. https://doi.org/10.1145/3397271.3401063. (2020)
31. Zhao K.Z., Wang X.T., Zhang Y.R., Zhao L., Liu Z., Xing C.X., Xie X.: Leveraging Demonstrations for Reinforcement Recommendation Reasoning over Knowledge Graphs. In SIGIR. 2020,239–248. (2020)
32. Xie Z.P., Jin C., Liu Y.: Personalized Knowledge Recommendation Model based on Constructivist Learning Theory. Journal of Computer Research and Development (in China),2018,55(1):125-138, Doi: 10.7544/issn100-1239. 201820160547. (2018)
33. Heffernan N.T., ASSISTments, 2014, https://sites.google.com/site/assistmentsdata/. (Accessed Dec. 29, 2021).

Yan Zhang received the B.S. degree in computer science and technology from Xidian University, Xi'an, China, in 2004 and the M.S. degree in computer application technology from Northwest A&F University, Xi'an, China, in 2011. His current research interests include machine learning, pattern recognition and network security.

Hongle Du received the B.S. degree in computer science and technology from Henan University, Kaifeng, China, in 2004 and the M.S. degree in computer application technology from Guangdong University of Technology, Guangzhou, China, in 2010. His current research interests include machine learning, pattern recognition and network security.

Lin Zhang is a professor in School of Mathematics and Computer Application, Shangluo University. He received the B.S. degree in computer science and technology from Shaanxi Normal University, Xi'an, China, in 1991. His current research interests include machine learning, Online learning and network security.

Received: July 06, 2022; Accepted: December 06, 2022.

Computer Science and Information Systems 20(2):879–892 https://doi.org/10.2298/CSIS220919017S

The Duration Threshold of Video Content Observation: An Experimental Investigation of Visual Perception Efficiency

Jianping Song[1], Tianran Tang[2], and Guosheng Hu[3*]

[1] School of Art and Design, Zhejiang A&F University,
Hangzhou 311300, China
[2] Department of Computer Engineering, Dongguan Polytechnic,
Dongguan 523808, China
[3] Design Innovation Center, China Academy of Art,
Hangzhou 310002, China
huguosheng@msn.com

Abstract. Visual perception principle of watching video is crucial in ensuring video works accurately and effectively grasped by audience. This article proposes an investigation into the efficiency of human visual perception on video clips considering exposure duration. The study focused on the correlation between the video shot duration and the subject's perception of visual content. The subjects' performances were captured as perceptual scores on the testing videos by watching time-regulated clips and taking questionnaire. The statistical results show that three-second duration for each video shot is necessary for audience to grasp the main visual information. The data also indicate gender differences in perceptual procedure and attention focus. The findings can help for manipulating clip length in video editing, both via AI tools and manually, maintaining perception efficiency as possible in limited duration. This method is significant for its structured experiment involving subjects' quantified performances, which is different from AI methods of unaccountable.

Keywords: Video Editing, Human Visual System, Perceived Efficiency, Footage Duration, Video Information.

1. Introduction

Along with the booming of computer vision technology in recent years, artificial intelligence (AI) techniques have been extensively applied in visual arts. Since IBM's Watson created the movie trailer for *Morgan* in 2016, AI application in video editing has become dramatically widespread and in-depth. However, video editing always requires an understanding of the perception mechanisms and laws of human vision system (HVS), no matter how advanced the editing technology is. The perceptual laws and principles of HVS is a key knowledge and prerequisite for ensuring that video works are correctly and effectively understood by audience. In fact, scientific researchers have

* Corresponding author

been focusing on the application of HVS mechanism in image processing since 1980s, and video artists are definitely the peer that started to focus on such issues earlier.

Video artists, especially directors and film editors, need to effectively manipulate the duration of each shots when dealing with the relationship between the narrative flow of videos and the audience's perception. Traditionally, artists rely on their visual experience and professional skills to fulfill their intentions. While, due to the widespread usage of computer vision tools of AI technology, it is urgently in need for definite quantification on such factors that directly affect the legible, definable and understandable of video works. Undoubtedly, clear quantitative laws can improve our understanding of watching behavior and be directly applied to AI-driven or manual video creation. Besides, these laws may benefit AI technology in recognition of video contents [1-3].

Technically, physiological and psychological fluctuation could rise on viewing the video contents, which intricately affects the audience's mastery and understanding of the video contents. Among them, the duration of the shots can directly affect the audience's perception of the video information. Video content is often made up of consecutive shots, each shot needs a reasonable length of exposure to meet the visual perception requirements. The audience will not have enough time to grasp the main content of the shot in inadequate duration, while lengthy duration will occupy the audience’s attention to the minor details of the scenes, thus distracting their attention from the main theme. In addition, audience’s visual perception of video is also influenced by the content of the images per se, and it should not be ignored that information density of video is also an important factor for cognitive efficiency.

This visual perception issue not only determines whether AI technology can meet the needs of audience-oriented video creation, but also is a technical problem that has long troubled video artists. To reveal this issue, it is necessary to analyze audience’s perceptual efficiency on various scenes through duration counted experiments. The experiment of this project statistically analyzed the efficiency and accuracy of audience's visual perception of video contents from the video editing perspective. With focusing on the perceptual perceptions of exposure durations and scene contents, the experiment aims to clarify the necessary shot duration for the audience to perceive and grasp the main information in the video.

Six video clips of different perspectives and contents were collected, and composed into five video clips of different durations as controlled experimental materials. Then, the subjects were divided into five groups according to the very video clip they watched and finished a questionnaire. Finally, the census data were analyzed and discussed according to the questionnaire results. Despite the fact that the experiment faced subjective factors, the results still shows a clear tendency. However, unexpected interference of the experimental results caused by factors, such as certain contents and gender differences, was also found.

This paper consists of six sections. Cognitive mechanism of video and multimedia are reviewed in section 2. Section 3 is about conducting the experiment that involves volunteers’ participation. The data of the experimental results are measured and analyzed in section 4, and discussion is conducted in section 5. Section 6 shows the summary of the conclusions.

2. Related Research

Rather than a complex optical imaging system, HVS is influenced by a variety of operating mechanisms that is often explained as psychological and neurophysiological issues, and is highly intelligent for information perception and processing. Marr, an expert in neurology and psychology, studied the visual system from the information processing perspective and proposed, for the first time, a sophisticated explanation of the basic construction of the visual system, which laid the theoretical foundation for the application of human visual cognitive mechanisms in image processing [4,5]. Since then, researchers have continued to improve the vision theory from multi-aspects, such as information acquisition, visual sensitivity, content perception, and gradually applied them in computer vision and image processing fields.

The work of this project concerned with visual attention and perception of video contents in terms of exposure duration, for the scene duration of video directly affects human access to the visual information. Johansson raised questions about the relationship between human perceptual speed and the moving objects [6]. He also studied perceptual organization on stimulus patterns, by simulating human motion as walking, running, etc., with ten moving bright spots, and tested human perceptual speed through different exposure durations of images [7]. Albright et al. made a further research on the visual perception of moving images, and suggested the importance of context factors in visual perception [8]. Chun investigated the interrelationship between context and visual attention, and the mechanism of contextual information learning and its guidance of visual attention deployment, suggesting that context cueing facilitates the efficiency of visual search and recognition [9]. With the booming of digital image technology and creation, the research on visual perception of images has extended in-depth. By investigation of the relationship between spatial frequency and image exposure duration, Watt affirmed the general features of human vision – turning from rough perception to detail perception as exposure duration increases [10-12].

The image perception efficiency of HVS is not only determined by the context, but also related to the familiarity of the theme and visual objects. Thorpe et al. investigated the influence of audience's familiarity with the theme and contents on perceptual efficiency, and showed negative significance of familiarity with the content when faced with complex perceptual objects [13]. However, Bülthoff and Newell's study showed that object recognition is directly related to familiarity with the contents. He claimed that recognition decisions are largely driven by familiarity [14]. Fabre-Thorpe et al. experimentally demonstrated that familiarity has significant effect on perceptual duration of simple images, but this indication is implicit in complex natural images [15]. They concluded that HVS relies on highly automatic feed-forward mechanisms in perceiving highly complex images, which are very little influenced by familiarity. While, categorical representations are crucial for memory load when visual content maintained for a certain duration [16]. This mechanism is also adopted in machine recognition of multimedia [17].

The other factor is the cognitive mode, precisely, the order of the perception process. In most situations, the attentional mechanisms of the visual system tend to coexist in both top-down and bottom-up modes. Posner and Petersen's study of the attentional properties showed that HVS pays attention to visual stimuli consciously when detecting signals for focal objects, and the consciousness is prominent in cognitive accounts of

attention [18]. Kastner and Ungerleider investigated the mechanisms of interaction between bottom-up and top-down modalities, and asserted that HVS has selective and active mental activity properties [19, 20]. Bar's study focused on and revealed the role of top-down perceptual mechanisms during low spatial frequency image perception [21]. In addition, the attentional mechanisms of HVS are strongly linked to emotional states to a large extent. Fan et al. confirmed the relationship between image attributes and audience's emotions by studying subjects' feedback on 31 imagery attributes [22]. Zhang et al. also insists that emotions play important role in video cognition [23].

We consider that these mechanisms and characteristics of HVS lie in the efficiency and accuracy of audience's perception of video information. If the temporal efficiency of video perception is analyzed quantitatively, the results will be contributive to improving the accuracy and efficiency of information dissemination in video art. This is the reason for conducting the experiment and clarifying the significance of this research work.

3. Experimental Design and Execution

In navigation perspective, there are usually two modes of visual perception: a top-down perception mode driven by subjective consciousness, in which the viewer intentionally intensify attention to certain visual information or actively search for confirmation of the expected information during the viewing process; the other is a bottom-up perceptual mode driven entirely by visual stimuli, in which the audience's perception depends on the stimulus element of the object itself. It usually occurs under conditions in which the viewer has no expectations or contextual preparation for the content and information, and perception is formed gradually from the process of visual stimuli [19, 24]. Top-down attention is mainly governed by consciousness, and visual perception actively seeks the target [25], while bottom-up attention depends primarily on the visual perceptual content. In most viewing experience, audience do not know what specific visual information they are about to access, and can only gradually form cognitive concepts by following the video narration, so it mainly belongs to the bottom-up perceptual mode. The perceptual mode set in this experiment is the bottom-up one, and the subjects participating in this experiment will not be informed of the relevant content before the test so as to maintain their ignorance of the visual information.

3.1. Testing Video and Questionnaire Design

In order to ensure the diversity of video types within the limited testing volume while restore the general viewing experience, we selected six video clips (Figure 1). The six scenes cover camera moving and still shots, camera perspectives ranging from overhead to elevation shots, close up to panoramic shots, with contents of characters, animals and vehicles moving, relatively still scenes with natural landscapes and architectural spaces, etc. (Table 1). Since each shot is prepared for various exposure duration, it is necessary to ensure that the motion of each video is continuously stable so as to avoid significant

information fluctuations. These clips are largely stable both in camera motion and target content, and can satisfy the process of arbitrarily cutting part of the motion.

Figure 1. The six clips of testing videos. The clips are intensively chosen according to the testing objects. Clip A and F are provided by https://www.vcg.com, and B, C, D, E from https://www.vjshi.com.

Table 1. Contents and features of the six testing clips.

Clip No.	*Camera Motion*	*Character and Content*	*Action*	*Testing Information*
A	Sideways dolly	city street, Woman, cellphone, tablet, watch, bag	walk, smiling, eyes shifting, holding the tablet	environment, woman, facial expression (smile), dress, hairstyle, bag, belt, cellphone, tablet, watch
B	still	grassland, sheep	sheep moving	grassland, wired fence, mountains, the white sheep, flock of sheep
C	Pilot view	roads, cars, buildings, trees	car running	running cars, roads, trees, buildings, roof view, cars in parking, electric bicycle, pedestrians, traffic signs on the ground
D	still	sky, plain, cattle, trees	almost still	blue sky, barren plain, trees, cattle and sheep, white mark on the ground, white cow foreground, aircraft trails in the sky
E	still	skyscrapers, buildings	cars running	main building, neighboring skyscarapers, buildings in the distance, traffic flow
F	following	street, two men, bicycle	walking nearer, talking	two men, tie, jacket on arm, walking forward, bicycles, briefcase

This experiment is based on materials of the same video clips. Since normal duration of a single clip is generally not less than one second, our test videos are set from one second on, with five gradient samples up to five second long. According to the experimental duration design, each original clip was cut for five samples, and five test videos were synthesized by same durations of every clip. The contents of each test video are the identical, but the segment durations are different, i.e., the one-second testing

video is stitched with the six original videos in one second clips, the two-second testing video with the original video in two-second clips, and so on to the five-second-long video. In order to facilitate testing operation, and avoid the montage effect formed by consecutive shots, which would affect the visual perception effect of the subject, three-second-long full-screen in medium-gray (128 of 256) were inserted between each two shots of the synthesized video to make intervals in between. The output size of these videos is 1920*1080 pixels with frame rate of 25p.

In order to evaluate the subjects' perception of the video content, information of the video was listed according to the detail levels, and a questionnaire was designed for the subjects to fill. The questionnaire provided three options for each listed object of the video contents: *definite*, *vague* and *null*. Subjects were asked to check one of the options based on their impressions after watching the testing videos.

3.2. Selection of Subjects

109 participants were invited for this experiment, all of them were university students of 18 to 25. Their academic majors are randomly scattered. In the original stage, 100 of the participants were divided into five groups, gender balanced, and each group watched only one test video of the specific duration, without repeating test for other videos (Figure 1). Some of the subjects did not finish all the test procedures as expected, and a few of them came to the results that were obviously not in line with common sense (possibly due to their reluctance to cooperate with the experiment). Their test data were excluded and other participants was asked to fill the gap. Finally, the number and gender ratio of subjects in each group were ensured to be comparable, with 100 questionnaires, half to half from each gender.

3.3. Experimental Environment and Equipment

The minimum difference in luminance that can be perceived by the human eye at a particular background is the luminance sensitivity level. According to Weber's law, the ratio of luminance sensitivity to background is commonly a constant. In an image of 256 gray levels, human vision tends to be highly sensitive to the areas with a gray level of about 128 [26, 27]. The six videos used in this experiment are all on luminance level of about 128, that guarantee the subjects better perceiving the video information. The brightness contrast of the video images is mainly in the medium. The entire experiment was conducted in a room without strong light interference, and the videos were played on a Dell 27-inch display (U2720Q) running on a PC with Windows OS. The subjects kept eyes about 70 cm away from the display. The video player program was Potplayer.

3.4. Testing Process

The experiment was operated and recorded by an operator. The operator first communicated with the subjects and informed them of the test procedure, but did not

reveal any information about the contents of the testing videos. The operator then started playing the video when the subject is ready. The video starts with a 5-second countdown sign, and then the first test scene is presented. The operator paused the video after each shot of the six scenes is played. The subjects were then given a questionnaire about the scene and was informed that the items in the questionnaire might not be presented in the video. The questionnaire was used to confirm what the subjects really perceived, with three options for each item: definite, vague, and not perceived. After finishing the questionnaire, the subject continued to watch the next scene, then the corresponding questionnaire, and all the video scenes were tested in this order. Figure 2 shows the work flow of the experiment.

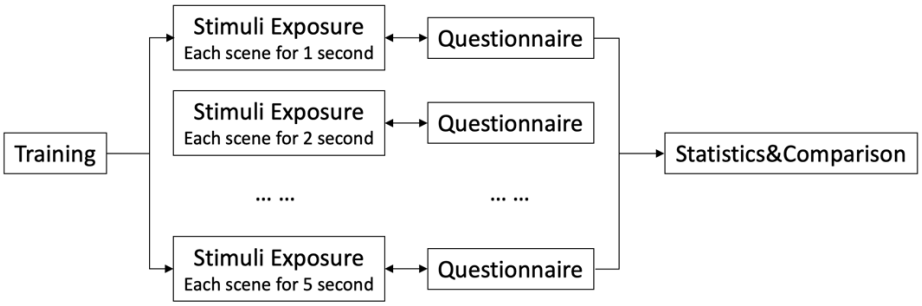


Figure 2. The work flow of the experiment. Each clip was inserted with gray background of three second between every scene. The subjects were asked to finished the questionnaire after watching each scene of the clip.

4. Measures for Data Validity

4.1. Design of the Test Videos

The themes of the six video clips were animals, human characters, buildings and nature spaces. These are common themes and typical types of general videos. In order to focus on the correlation of visual perception and exposure duration, we intentionally typified the video content when selecting the video clips and ensured that the motion trajectories of the main objects in the videos were conservatively stable, and the motion of the character objects was controlled within 1°/sec. This ensures both the typicality of the videos and the stability of the visual information when cutting shots for various durations.

HVS usually do not fully grasp the visual signals entering the human eye, there is but an Internal Generative Mechanism to interpret the input visual signals [28]. HVS will derive and predict the visual contents of the scenes to be recognized mainly based on the memory and experience, while the unintelligible and uncertain information will be

discarded. From this perspective, our experiment is about the HVS efficiency of visual information derivation. The efficiency is influenced by human memory. For instance, people are familiar with scenes viewed from a normal viewpoint, as opposed to unfamiliar things, so that people perceive differently in things of familiarities [14]. The overhead perspective is not a common observation angle, so there will be little prior memory information. For this reason, we deliberately chose an aerial video of an overhead view, hoping to access whether the amount of visual experience make differences in visual perception by comparing with other items.

HVS is inevitably influenced by complex factors when perceiving content. Prior contextual information constrains anticipation and visual navigation, facilitating the recognition and search for objects in complex images [29]. To avoid such contextual cues, we inserted a 3-second null screen between each scene to discard consecutive shot combinations that might form extended interpretation of primitive contents, and also to facilitate the pause operations during the experiment.

4.2. Design of the Test Questionnaire

The test questionnaire for this experiment underwent two major versions during the design phase. The original questionnaire included open questions, asking the subjects what they saw in the footage that had just been shown. We found that this approach would result in the subjects missing lots of important information that he actually perceived, and cause uncertainty in the subjects' descriptions that could not be translated into countable data. Thus, we redesigned the questionnaire to list all the content that appear in the videos for the subjects to check off the items that they perceived. However, the results of this test also showed significant inaccuracies, as the subjects appeared irresolute on some of the content and randomly check the items on their uncertainty. It indicates that the results would be seriously affected by the fact that subjects with this level of perception when filling out the questionnaire. Finally, we decided to provide three options for each item: *definite*, *vague*, and *null*. Only those items checked for *definite* were counted as the positive results, so as to filter out some ambiguous perceptions and improve the accuracy of the information perceived with certainty (Table 2).

Table 2. Questionnaire for Scene D of the testing clips (executed in Chinese version). Subjects were asked to check the perceived items according to their impression.

	blue sky and earth	*trees*	*cattle and sheep*	*white mark on the ground*	*white cow in the foreground*	*aircraft trace in the sky*
definite	√		√		√	
vague		√		√		
null (missed)						√

4.3. Plan of the Testing Process

The major challenge of this experiment is ensuring the accuracy of the result data, since visual perception is easily affected by complex factors, such as empirical information, experimental environment, and individual differences. Theoretically, having each subject check all the test videos of various durations can eliminate bias of the individual subjects, but the results would be obviously affected by visual memory and contextual cues when the subjects were repeatedly exposed to the videos of identical contents. Therefore, we decided to have each subject watch every video only once regardless the duration. Even though the test results were still inevitably influenced by individual differences, mental state, and attitudes of the subjects when filling out the questionnaire. In order to reduce these influences, we specifically limited the test subjects to university students, who are of similar age, mental state, cognitive ability and life experience, and therefore have relatively minor individual differences. The test was conducted in their spare time for relax and attentive mood. Nevertheless, many problems were still found in the statistics of the test data. In particular, for one-second duration test, some questionnaires ticked all the listed items as *definite*. As a matter of fact, HVS is unlikely to grab all the information in such a short time. Considering that these questionnaires would weaken the significance of the statistical results, we eliminated the answer sheets with all the items checked as *definite*, and invited more subjects to participate in the questionnaire for the absence.

5. Results and Discussion

The perceptual procedure of human vision usually starts from the overall observation of the imagery, then quickly locates the important targets for in-depth examination. The visual attention actively focuses on the targets of interest, while selectively reducing attention or even ignoring other uninteresting objects. This active and selective observation behavior is also called the *visual attention* mechanism. Imagery in the vision usually contains complex visual information consisting of various objects juxtaposed or superimposed, rather than a single figure, which interact or interfere with each other causing *visual masking effect* as Macknik referred to [30]. Some information is dominant in the interrelations, showing active aggressiveness, while others are subsidiary, negative or weak. Thus, the objects in motion pictures often present a very rich hierarchy of strengths and weaknesses due to visual perception mechanisms. Video information can be divided into three categories according to the visual perceptual strength: the global information, the major objects of attention, and the subsidiary objects. Taking the perception of these three information categories into account, it is assumed that the order of human visual perception is from the global information to the major objects and then expands to the subsidiary objects, where the global and major categories of information are usually clearer and more explicit in the audience's visual perception, while the subsidiary information is often intentionally weakened by the creator to be vague in the audience's perception. Therefore, when providing the three questionnaire options of *definite*, *vague* and *null*, the subsidiary information was intentionally filtered out. The target of the survey was focused on the main objects of the

video artists' emphasis. The results of the questionnaires in which the subjects chose the *definite* option remarkablely showed this tendency, i.e., the main information of the scenes were mostly found in the *definite* option (Figure 3).

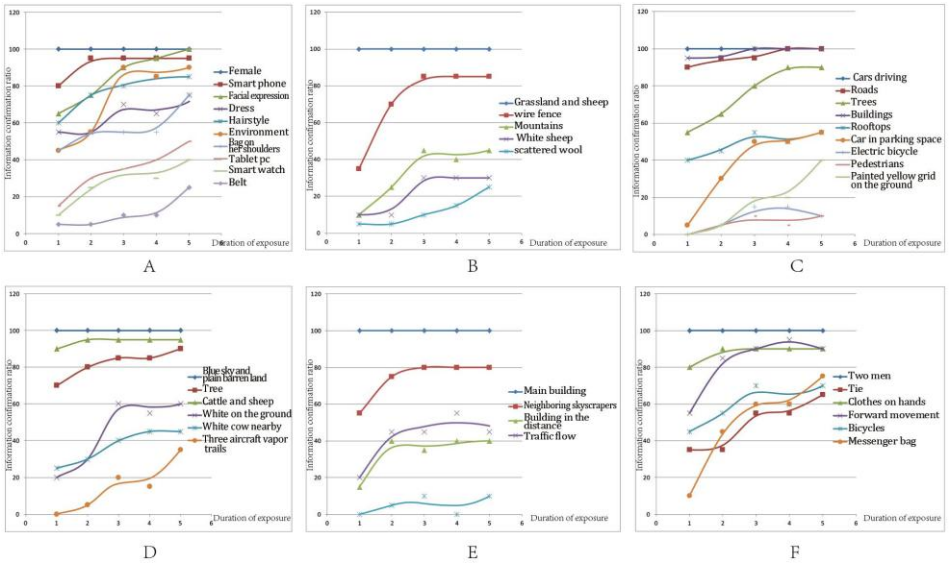


Figure 3. Statistics of the cognitive trends according to the audience questionnaire on the six-scene test. The major the content, the higher the confirmation rate shows; and the longer the exposure duration, the higher the confirmation rate.

The statistic trend shows that as the perceived time increases, more items is confirmed by the subjects (Figure 3). Among them, the global information in the uppermost layer is mostly grasped on one-second duration. The information of main objects is most sensitive to the perceived duration, with some major details in the upper level in the graph and others in the middle level. The perception curves of most major details show significant surge between two and three seconds of exposure duration, and leveling off after three seconds. Among them, the trend curves of scene *E* come to the high stage period significantly early. It is easy to notice, by inspecting the content of the video and the statistic curves, that the content of scene *E* are more concise and has less major details. It is very likely the main reason that supports the rapid achievement of higher perceptual certainty. Therefore, it is the detailed information of main objects that is most obviously affected by perceived duration. As the shot duration increases, the perceptual certainty of the characters and primary details increases significantly. The perceptual certainty of the subsidiary details also increases along the increase of perceived duration, but with a certain lag comparing to the characters and primary information, and the increase is relatively smoother. They are mainly unimportant details with weak attraction, many of them are even completely ignored by the subjects. However, most information of these items does not affect the subject's understanding of the video. In general, the statistical results of the test data seem in consonance with the

visual attention mechanism, and reveal the perceptual order and attention distribution of the three levels of visual information.

In this experiment, the subjects were divided into two identical groups of both genders, in order to eliminate gender bias, and also for convenience of inspecting the gender differences. Judging from the features of the questionnaire data, the gender difference is significantly reflected in the test of scene *A* (Figure 4). The statistical results of the questionnaire on scene *A* show that female subjects pay more attention to the character's dress, bag, hair style and belt in the video than the male group, and the intensity and certainty of perception increases earlier. Whereas male subjects tend to be much acute on electronic items as the tablet, cellphone and smartwatch. There seems to be a lack of attention to the women character's belt. For scene *D*, gender differences had less impact on the perception of global information, and both male and female subjects were able to identify the grass and sheep in the picture in just one second. The barbed wire fence belongs to the second category of information, i.e., the major object, which was fully perceived with perception up to three seconds, but the subsidiary information did not significantly increase in perceptual intensity within all the testing durations (Figure 5). In general, gender differences do exist in visual perception and are likely to be influenced primarily by interest and culture.

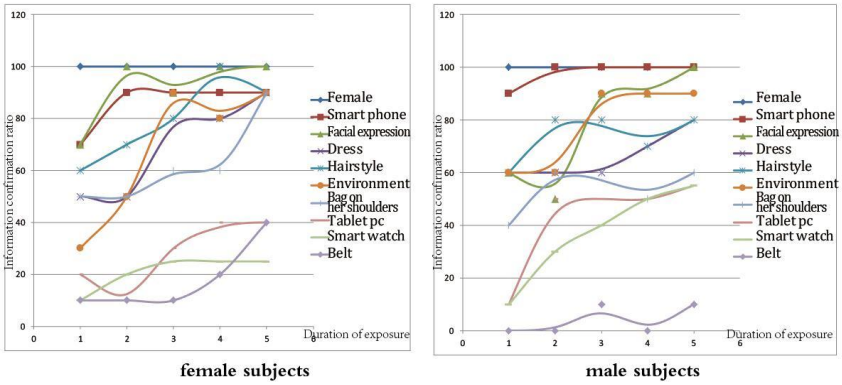


Figure 4. Gender difference trend according questionnaire on scene A.

An interesting phenomenon appeared that most male subjects showed a significant surge of awareness of the frost on the ground in scene *D* at the third second. There were also perceptual differences on the white cow, with more males than females definitely grasped it in one second, but this advantage did not increase as the exposure duration increased, while the number of females who positively checked this information increased significantly. This may imply that there are differences in the order of visual perception, not only between genders, but also between individuals of the same gender.

Besides, there shows no significant difference, merely by the results of the questionnaire, in the video duration of both normal viewpoints and unconventional perspective, such as aerial footage.

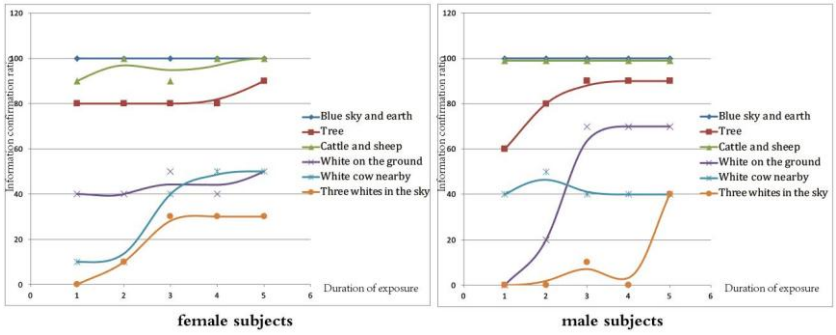


Figure 5. Gender difference trend according questionnaire on scene D.

6. Conclusion

The experimental results are basically consistent with our initial hypothesis, but the specific data fluctuate according to the video content. It generally shows that three seconds is a very important threshold. HVS can perceive the overall contextual information and the main character-objects in videos within exposure duration of three seconds. In fact, once these two types of information are mastered, the content and intention of the video can be basically understood. This indicates that audience can grasp the main information of video shot from three seconds up. However, this perceived duration fluctuates due to the information density of the footage. In the case of relatively low information density, the shot duration can be as short as two seconds, but in most cases segment duration of three seconds or longer appears to be more secure. Gender differences also affect the perception of certain types of information, such differences depend mainly on the audience's interests rather than on the visual perception mechanism per se.

Although the experimental results are generally consistent with our hypothesis, there are still shortcomings that affect the accuracy of the experimental results. First, the greatest impact on the accuracy of the experimental results might be the subjective factors of the questionnaire. Although we screened the received questionnaires, it is still hard to say the retained questionnaires were sufficiently objective. The other uncertainty is that the testing process was conducted simultaneously by two testing operators. Although we trained the operators before the test and controlled the testing process equally, the collaboration differences between the two operators and subjects were still unavoidable and could affect the test results. In addition, this experiment only selected six typical videos for testing, it might be inadequate to cover the miscellaneous and complex perceptual cases. Plenty of factors on video perception needs to be investigated further

Acknowledgment. This document is the results of research project supported by The National Social Science Fund of China [20ZD09] and China Academy of Art [ZD202205].

Data Availability. The raw data supporting the conclusions of this article will be made available by the authors by email or website.

References

1. Gkalelis, N., Goulas, A., Galanopoulos, D., Mezaris, V. Object Graphs: Using Objects and a Graph Convolutional Network for the Bottom-up Recognition and Explanation of Events in Video. Computer Vision and Pattern Recognition, IEEE. (2021)
2. Schwenzow, J., Hartmann, J., Schikowsky, A., Heitmann, M. Understanding videos at scale: How to extract insights for business research. Journal of Business Research 123:367-379. (2021)
3. Zhang, J., Yu, X., Lei, X., Wu, C. A Novel Deep LeNet-5 Convolutional Neural Network Model for Image Recognition. Computer Science and Information Systems 19(3):1463-1480. (2022)
4. Conrad, M., Cin, M.D., Marr, D. Approaches to biological information processing. Science 190:875-876. (1975)
5. Marr, D. Early processing of visual information, Philosophical Transactions of the Royal Society of London. Biological Sciences 275(942):483-519. (1976)
6. Johansson, G. Visual perception of biological motion and a model for its analysis. Atten. Percept. Psychophys 14:201-211. (1973)
7. Johansson, G. Spatio-temporal differentiation and integration in visual motion perception. Psychological Research 38(4):379-393. (1976)
8. Albright, T.D., Stoner, G.R. Visual motion perception. Proceedings of the National Academy of Sciences of the United States of America 92(7):2433-2440. (1995)
9. Chun, M.M. Contextual cueing of visual attention. Trends Cognit Sci, 4(5):0-178. (2000)
10. Watt, R.J. Scanning from coarse to fine spatial scales in the human visual system after the onset of a stimulus. Journal of the Optical Society of America A-optics Image Science & Vision 4(10):2006-2021. (1987)
11. Bicanski A., Burgess, N. A Computational Model of Visual Recognition Memory via Grid Cells. Current Biology 29(6):979-990. (2019)
12. Rybak, I.A., Golovan, A.V., Gusakova, V.I. Behavioral model of visual perception and recognition. Proceedings of SPIE - The International Society for Optical Engineering 1913:548-560. (1993)
13. Thorpe, S., Fize, D., Marlot, C. Speed of processing in the human visual system. Nature 381:520-522. (1996)
14. uBülthoff, I., Newell, F.N. The role of familiarity in the recognition of static and dynamic objects. Progress in Brain Research 154:315-325. (2006)
15. Fabre-Thorpe, M., Delorme, A., Marlot, C., Thorpe, S. A limit to the speed of processing in ultra-rapid visual categorization of novel natural scenes. Journal of Cognitive Neuroscience 13(2):171-180. (2001)
16. Zhou, C., Lorist, M.M., Mathôt, S. Categorical bias as a crucial parameter in visual working memory: The effect of memory load and retention interval. Cortex 154:311-321. (2022)
17. Zafar, B., Ashraf, R., Ali, N., Ahmed, M., Jabbar, S., Naseer, K., Ahmad, A., Jeon, G. Intelligent Image Classification-Based on Spatial Weighted Histograms of Concentric Circles. Computer Science and Information Systerms 15(3):615-633. (2018)
18. Posner, M.I., Petersen, S.E. The attention system of the human brain. Annual Review of Neuroscience 13(1):25-42. (1990)
19. Kastner, S., Ungerleider, L.G. Mechanisms of visual attention in the human cortex. Annual review of neuroscience 23:315-341. (2000)

20. Intraub, H. The representation of visual scenes. Trends in Cognitive Sciences 1(6):0-222. (1997)
21. Bar, M., Kassam, K.S., Ghuman, A.S., Boshyan, J., Schmidt, A.M., Dale, A.M., Hamalainen, M.S., Marinkovic, K., Schacter, D.L., Rosen, B.R., Halgren, E. Top-down facilitation of visual recognition. Proc Natl Acad Sci USA 103(2):449-54. (2006)
22. Fan, S., Koenig, B.L., Zhao, Q., Kankanhalli, M.S. A Deeper Look at Human Visual Perception of Images. SN Computer Science 1(1):58. (2020)
23. Zhang, J., Wen, X., Whang, M. Recognition of Emotion According to the Physical Elements of the Video. Sensors 20(3):648. (2020)
24. Privitera, C.M., Stark, L.W. Algorithms for defining visual region-of-interesting: comparison with eye fixations. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(9):970-982. (2000)
25. Baluch, F., Itti, L. Mechanisms of top-down attention. Trends in Neurosciences 34(4):210-224. (2011)
26. Netravali, A.N., Haskell, B.G. Digital Pictures: Representation and Compression. New York: Plenum (1988)
27. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing 13(4):600-612. (2004)
28. Peter, R.J., Iyer, A., Koch, C., Itti, L. Components of bottom-up gaze allocation in natural scenes. J. Vison 5(8):692-692. (2005)
29. Joubert, O.R., Rousselet, G.A., Fize, D., Fabre-Thorpe, M. Processing scene context: fast categorization and object interference. Vision Research 47:3286-3297. (2007)
30. Macknik, S.L., Livingstone, M.S. Neuronal correlates of visibility and invisibility in the primate visual system. Nature Neuroscience 1(2):144-149. (1998)

Jianping Song, lecturer of digital media art at Zhejiang Agriculture and Forestry University, is mainly engaged in teaching and research in the field of digital graphics and videos. His interests are in the human visual perception of mages, especially from the perspective of video creation and artificial intelligence.

Tianran Tang, PhD of Design, associate Professor, is the deputy dean of Artificial Intelligence College of Dongguan Polytechnic. His main interest is in teaching and research of digital art and animation design, including the promotion of digital art and animation brands.

Guosheng Hu, MA in art and design, PhD in computer technology, is an associate professor of China Academy of Art. His work and research interests are related to computer graphics, color design, and multimedia & interaction art.

Received: September 19, 2022; Accepted: January 16, 2023.