Computer Science and Information Systems 21(4):1845–1864 https://doi.org/10.2298/CSIS240418059W

Multi-object Real-time Tracking for Intelligent Breeding of Animal

Fei Wang[1], Bin Xia[2,⋆], and Liwu Pan[1]

[1] School of Information Engineering, Henan University of Animal Husbandry Economy
Zhengzhou 450044 China
feinix@126.com
[2] Zhengzhou Cotton&Jute Engineering Technology and Design Research Institute, All China Federation of Supply and Marketing Cooperatives
Zhengzhou 450044 China
xiabinwto@163.com

Abstract. Animal intelligent breeding utilizes advanced technology and intelligent systems to monitor, analyze, and optimize animal growth environments and management, which can enhance breeding efficiency and animal health levels. In this paper, we propose a new multi-object real-time tracking within deep framework for intelligent breeding of animal (MRT-IB), which consists of semantic feature extraction module, center point prediction module, and object and trajectory calibration module. MRT-IB reduces the difficulty of modeling animal trajectories by performing animal detection on consecutive frames, resulting in higher robustness in real farming scenarios compared to traditional multi-object tracking schemes that directly model animal motion trajectories.

Keywords: animal intelligent breeding, multi-object real-time tracking, object and trajectory calibration.

1. Introduction

Intelligent breeding is an innovative approach that integrates advanced technology with animal husbandry practices [4,21,19]. It improves farming efficiency and animal welfare by continuously and automatically monitoring animal behavior and taking timely and appropriate measures when anomalies are detected. Intelligent breeding technology is supported by advanced artificial intelligence algorithms and hardware systems, including various sensors such as microphones, cameras, accelerometers, and global positioning systems. These sensors continuously collect biological response data from animals, such as sound, behavior, movement, and location. Then, by analyzing these data and extracting features relevant to target behaviors, machine learning models are trained to achieve real-time detection, tracking, and monitoring of abnormal animal behavior [17,22,8]. Once the system detects abnormal behavior or health issues, it immediately issues alerts and takes corresponding actions, such as alerting veterinarians or automatically adjusting the animals' environmental conditions. This real-time monitoring and intervention help farm managers promptly identify and address issues, thereby improving animal welfare and production efficiency.

⋆ Corresponding Author

In recent years, with the rapid development and widespread application of deep neural networks, numerous real-time animal monitoring algorithms based on multi-object tracking have been proposed for smart farming. Ahrendt et al. designed a tracking system for free-range animals, using partial animal images to establish a 5D Gaussian model for individual animal targets. This system can track at least 3 animals over long time spans [1]. Martin et. al proposed a multi-object tracking method based on joint kernel correlation filtering. They designed a joint filter based on a pure target appearance model of kernel correlation filtering, effectively addressing the problem of poor algorithm performance in cases of severe target occlusion and complex environmental backgrounds [25]. Matthews et al. proposed a multi-object tracking algorithm using the Hungarian algorithm to match detection results from adjacent frames. The algorithm performed well in detecting individual targets and tracking consecutive frames in 3D video sequences of group-raised pigs, achieving an overall tracking accuracy of 89% for pig targets [18]. Van der Zande proposed a multi-object tracking algorithm based on optical flow and interframe interpolation for tracking dairy cows during rumination, achieving a final accuracy of 89.12% [27]. Philipp et al. proposed a multi-object tracking algorithm based on multi-feature detection and target association. They used color and texture features for adaptive detection of animals, edge features for image segmentation, and the Kuhn-Munkres algorithm to determine the association between animal targets and their mask image regions. Experimental results showed that the algorithm has strong robustness and meets real-time tracking requirements for multi-object tracking [5]. Zhang et al. combined a CNN-based object detection model with a deep correlation filtering tracking algorithm to track all pig targets' trajectory coordinates in structured pigsty environments. Test results showed that the tracking accuracy reached 94.72% [28]. Ali et al. proposed a multi-object tracking algorithm based on object detection, locating the positions of detected targets in single-image video frames, and then connecting the trajectories of detected targets in consecutive frames. Experimental results showed that the best tracking accuracy of 92.68% was achieved with a synchronous search for subsequent frames of 9 frames and a tracking duration of 30 frames [2].

In current research combining computer vision technology with the animal husbandry, researchers mostly focus on introducing computer vision technology to address specific problems within the farming process. This approach often yields relatively good results. However, experiments in these studies often require artificially imposed constraints. For instance, in the detection of hoof problems in cattle, researchers limit the number and poses of cattle in a video segment. They then detect, record, and calculate the frequency of each cow's hoof strikes to infer whether the cow is lame based on empirical data. While this method can achieve high detection accuracy, the manual recording of identities for each detected cow increases the burden on farming personnel. Many studies also attempt to use detection networks to simultaneously detect the behavior of multiple animals in images, aiming to improve farming accuracy by monitoring the daily status of livestock. These efforts often yield highly accurate behavior detection results. However, since these methods cannot distinguish between different animals within a video segment, they can only provide the number of animals exhibiting a particular behavior at a given time, without capturing the specific behavioral changes of individual animals over time. Therefore, further research is needed in current computer vision technology applied in the farming

industry to continuously identify and differentiate between different animals over a period of time, in order to obtain more precise records of animal status information.

To this end, we propose a new multi-object real-time tracking based on feature consistency for intelligent breeding of animal (MRT-IB), which comprises three parts, i.e., semantic feature extraction module, center point prediction module, and object and trajectory calibration module. Specifically, In the tracking task, MRT-IB considers the similarity of theposition and appearance of the same animal between consecutive frames.and the differences from other animals, and proposes a multi-objecttracking scheme that uses animal detection results and animal position and appearance semantic features as similarity criteria. Based on this, acenter point prediction module is designed to use the continuity of animalposition in the video to assist in detection and tracking. This schemereduces the difficulty of modeling animal trajectories by performinganimal detection on consecutive frames,and is more robust thantraditional multi-object tracking schemes that directly model animaltrajectories in real farm scenes. At the same time, experimental results onmultiple farm scene datasets demonstrate that the center point predictionmodule can effectively improve the detection and tracking accuracy ofmany similar animals appearing in the same scene. Finally, a plethora of experiments validate the effectiveness of MRT-IB in the intelligent breeding of animal.

Section 2 reviews existing smart farming methodologies, particularly focusing on computer vision techniques for livestock monitoring. Section 3 introduces MAR-IB (Multi-Animal Recognition with Individual Behavior) and its methodology for addressing individual animal identification challenges in video streams. Section 4 presents experimental results evaluating MAR-IB's performance in real-world farming scenarios. Finally, Section 5 draws conclusions on MAR-IB's effectiveness and implications for livestock management in smart farming.

2. Related Works

In this section, we will introduce researches on computer vision technology and livestock intelligent breeding.

2.1. Computer vision technology

Computer vision technology is an extremely important branch of artificial intelligence, and it can now be divided into three most basic tasks: Image Classification, Object Detection, and Image Segmentation. In response to different task scenarios and requirements, research in computer vision has continuously refined, leading to the emergence of tasks such as Multiple Object Tracking (MOT) and Action Recognition [11].

For image classification algorithms, traditional methods such as KNN, SVM, and the Mixture of Gaussians model dominated the field before 2012, offering better performance in classification tasks. However, after the neural network-based AlexNet achieved a performance far exceeding traditional algorithms in the ImageNet LSVRC-2012, neural networks became the mainstream in the field of image classification [15]. Subsequent models like VGGNet, GoogLeNet, ResNet, and EfficientNet have continuously pushed the upper limits of classification task performance [21]. Recently, classifiers based on Transformers have reached the state of the art on major public datasets, becoming a new research

hotspot. Regarding object detection algorithms, the current mainstream algorithms are all based on deep learning. Object detection algorithms can be divided into two-stage object detection algorithms and one-stage object detection algorithms based on the algorithm's phase. Two-stage object detection algorithms require an additional Region Proposal Network (RPN) module; the most classic algorithms in this category are the R-CNN series, such as Fast R-CNN and Mask R-CNN [19]. One-stage object detection algorithms do not have an extra region proposal module and are trained end-to-end. A highly classic algorithm in this category is the YOLO series, including YOLOv3 and YOLOv4. Other algorithms like SSD, CenterNet, M2Det, and EfficientDet also have their innovative aspects [22]. Image segmentation algorithms can be divided into two major categories: Semantic Segmentation and Instance Segmentation. Semantic segmentation only requires the segmentation of objects of different types without distinguishing between different individuals of the same category. Classic algorithms include FCN, U-Net, PSPNet, DeepLab v3 Plus [8]. Instance segmentation, on the other hand, needs to differentiate between different individuals of the same category. Currently, the most popular algorithms are Mask R-CNN, SOLO v2 [23].

Multi-Object Tracking (MOT) algorithms can be divided into two subtasks: object detection and data association. Object detection algorithms are used to identify the objects that need to be tracked; data association algorithms assign identity identifiers (IDs) to the detected objects, ensuring that the same object is assigned the same ID across different image frames. These two tasks can be completed by a single network or by two independent networks in cascade. Based on the way these two subtasks are completed, the current mainstream deep learning-based multi-object tracking algorithms can be divided into two categories: single-step MOT and two-step MOT. Single-step MOT uses a single network to complete the detection and tracking tasks of objects, usually outputting information such as category, bounding box, ID, etc., through a detection head, for example, FairMOT, which has achieved high performance on multiple datasets [24]. Two-step MOT separates object detection and data association, so the actual focus of the work is on the data association task, such as Sort, DeepSort, and MOTDT, graph neural network algorithms [9]. The final implementation of the algorithm can be completed through the cascade of a detector and an associator, for example, the classic YOLOv4+Deepsort algorithm. Some algorithms delegate most of the work to the detector, such as Trackot++ proposed by Philipp Bergmann et al., pushing the limit of detection-based tracking [5]. Action recognition algorithms, in a narrow sense, only require the classification of individual videos or sequential images, such as the classic Two-Stream algorithm. However, in a broader sense, action recognition algorithms can be further divided into temporal action detection algorithms and spatiotemporal action detection algorithms. These corresponding tasks are more challenging, as they require the analysis and processing of long videos and can actually be considered a part of video content understanding. Among the most classic algorithms in this field are R-C3D, BMN, and TSP [3].

2.2. Livestock intelligent breeding

Current research combining livestock farming with computer vision often focuses on specific scenarios, where researchers apply various deep neural networks to the study of animal husbandry for multiple purposes within the breeding process. Object detection networks are primarily used by researchers for the detection of specific animals, such as

pigs, cattle, and sheep. There are also studies that apply detection networks to the identification of animal abnormalities. For example, Kang et al. developed a dairy cow limp scoring system, using a deep learning-based object detection network for the localization of cow hooves in videos, with an average precision of 87.0% [13]. The study then used the detected positions of cow hooves from consecutive frames as input, calculating the difference in lift-off and ground contact times of a specific cow's hooves to determine if the cow was limping. Cermek et al. used YOLOv2 on RGB images to detect the occurrence of digital dermatitis in cattle through digital image detection, achieving an accuracy rate of 88% [6]. Yang et al. employed Faster R-CNN and extracted the occupation index (the ratio of the head to the feeding area) as a spatial feature to identify the feeding behavior of multiple pigs [26]. Chen et al. used Inception and LSTM to extract spatiotemporal features for identifying pig feeding behavior at the group level [7]. This study proposed an image processing algorithm based on maximum entropy segmentation, HSV (hue, saturation, and value) color space transformation, and template matching, calculating the roundness of the head, the ratio of the head to the feeding area, the cumulative pixel count of head movement, and the distance from the head to the numbered pig's back to determine the identity and feeding time of each pig. These studies demonstrate the great potential of computer vision systems in identifying dairy cows with digital dermatitis, reducing the incidence of skin disease, and improving animal welfare.In addition to using deep neural algorithms to directly achieve research objectives, some studies combine deep neural networks with other algorithms in an attempt to achieve better results. For example, Lee et al. proposed the use of deep learning methods to remove backgrounds or detect objects in order to eliminate unnecessary noise and thus enable better post-processing analysisn [16]. They first processed video frames using some image enhancement methods, such as treating moving frames in the video with a Gaussian mixture model. Subsequently, they utilized TinyYOLOv3 to detect individual pigs in each selected frame, and then segmented the detected pigs. By using automatic image thresholding to complete the size calculation of the pigs, they achieved better results than simply using a segmentation detection model like Mask R-CNN [12]. These methods, which do not employ end-to-end deep learning approaches, often have better detection accuracy than their corresponding end-to-end deep learning algorithms by proposing hybrid models like the one mentioned above. Regarding the application of tracking technology in the breeding industry, since traditional computer vision tracking technology mainly relies on trajectory fitting to track detected objects, in animal husbandry, researchers primarily complete tracking tasks by directly contacting the breeding animals. For instance, distinguishing different animals by applying pigmented markings on their bodies, tracking them with the human eye, and then recording the status of animals with different markings at different time points [14]. Alternatively, animals are fitted with electronic devices such as ear tags based on RFID technology, and signals are received by a receiver to obtain information corresponding to different animals. With the increasing concern for animal welfare in recent years, and considering that methods like ear tags, which involve direct contact with animals, can cause injury during the contact process or be damaged during the animal's movement, researchers are also looking for ways to complete tracking tasks without contacting the animals. Su et al. applied a single-object tracking network based on the Siamese mechanism to track individual goats in surveillance videos, successfully tracking a specific goat even when multiple goats appeared in the same video segment [20]. Gan et al. applied the

Deepsort, a multi-object tracking framework based on detection, to the task of tracking multiple pigs in a pen. By leveraging the prior knowledge that the number of pigs in a pen is constant, they introduced a trajectory state correction mechanism to build a real-time pig tracking system [10]. Although deep learning-based tracking frameworks emerged relatively late compared to object detection frameworks, and the difficulty of data annotation is high, there is limited work combining livestock farming with single-object and multi-object tracking algorithms. However, the existing work indicates that tracking tasks have significant application scenarios in animal husbandry. They can effectively handle real-time tracking tasks for specific individuals or groups of animals, improving the efficiency of breeders and reducing their workload.

The widespread application of computer vision research in the field of animal husbandry primarily focuses on utilizing anchor-based detection architectures to identify animals in specific scenes within a single frame of an image. This approach requires researchers to manually adjust anchor parameters according to different scenes to accommodate changes in the environment and does not efficiently analyze detection results over time down to the level of individual animals. Additionally, detection networks tend to focus excessively on extracting features related to the current frame, neglecting the continuous information present in video frames, which can be used to assist in detection and tracking. These issues are also the focus of the research in this paper. The reliance on anchor-based detection architectures, while effective in many cases, can be limiting when it comes to adapting to dynamic environments and varying animal behaviors. The need for manual tuning of anchor parameters can be labor-intensive and may not always result in optimal detection performance, especially when there are significant changes in the scene or when the lighting conditions vary. Moreover, the focus on per-frame feature extraction can lead to the loss of valuable temporal information that is inherent in video data. This continuous information can provide context that helps in understanding the behavior and movement patterns of animals over time, which is crucial for accurate tracking and identification of individual animals. To address these challenges, the research presented in this paper aims to explore alternative approaches that can better leverage the temporal dimension of video data, potentially improving the accuracy and efficiency of animal detection and tracking in various farming scenarios. By incorporating methods that can capture and utilize the temporal coherence between frames, the goal is to develop a more robust and automated system for animal monitoring in the context of livestock farming.

3. Multi-object Real-time Tracking for Intelligent Breeding of Animal

In this section, a new multi-object real-time tracking based on feature consistency is proposed for intelligent breeding of animal (MRT-IB), which comprises three parts, i.e., semantic feature extraction module, center point prediction module, and object and trajectory calibration module. The main mathematical notations used in the paper are listed in Table 1.

3.1. Semantic feature extraction module

DLA-34 network is composed of multiple modules, each of which includes a series of convolutional layers and residual connections. It is capable of fusing feature maps from

Table 1. Frequently used notations

Notations	Description
$f(\cdot)$	the feature extractor
v^t	the t-th video frame in the breeding farm
z^t	the deep semantic representation of v^t
H^t	the heatmap spatiotemporal feature
$\bar{H}^t$	the predicted spatiotemporal feature
P^t	the predicted center heatmap P^t
$\bar{z}^t$	the fusion representations $\bar{z}^t$
a_i	an animal in the t-th video frame
τ_i	the trajectory of the animal a_i
e_i	feature vector of the animal a_i
d_{ij}	the similarity between the i-th animal in the current frame and the j-th active trajectory in the current system.
L_k	the detection box loss
α,β	the balance coefficients

shallow and deep layers, allowing the network to simultaneously learn high-level semantic information and low-level detail information. This aggregation of multi-scale features helps the network to better understand the content of images, especially when dealing with complex structures and diverse textures in the environment of breeding farms.

Therefore, MRT-IB utilizes the DLA-34, which has been pre-trained on the COCO detection dataset, as the feature extractor $f(\cdot)$ to learn deep semantic representations of video frames. Specifically, given the t-th video frame v^t in the breeding farm, the deep semantic representation z^t is obtained via:

$$z^t = f(v^t) \tag{1}$$

3.2. Center point prediction module

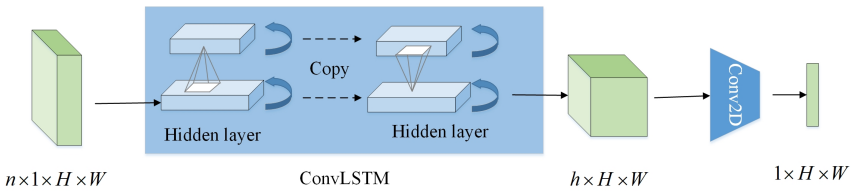

Fig. 1. The structure of center point prediction

The appearance and position of each object in surveillance videos exhibit a certain degree of continuity, which can be leveraged to assist detection in the current frame. Many related works in video detection and multi-object tracking have attempted to model the similarity of object appearances across consecutive frames using Siamese networks or attention mechanisms to enhance the network's detection capabilities in the current frame. This paper optimizes the complex process of feature similarity learning into directly predicting the positions of each object in the current frame. Additionally, an anchor-free

architecture can utilize the center layer to directly detect the position of the object. Therefore, this paper designs the center point prediction module to generate the center points of the current frame's center layer based on the output of the previous frame, as shown in the fig. 1.

Specifically, the center point prediction module first needs to dimensionally expand the centers of the previous n frames. Then, by concatenating along the expanded dimensions, a heatmap spatiotemporal tensor H^t with dimensions of $n \times 1 \times H \times W$ is obtained, which contains information about the changing positions of various animals in the video over time. In surveillance videos, if the animals are far apart from each other, the movement trajectories of each animal can be considered as independent trajectories. When animals are in close proximity, due to the constraints of their volume, the trajectories of the animals may have spatial relative relationships that could potentially influence each other. To capture these spatial relative relationships, this study considers using convolutional kernels for acquisition. Therefore, the center point prediction module selects ConvLSTM to process the animal heatmap spatiotemporal tensor, which can extract spatial features and perform temporal feature processing through h hidden layers, thus addressing the length and width of the current heatmap. ConvLSTM first performs h spatiotemporal convolution operations to extract the spatiotemporal hidden layer features of the movement trajectories of each animal within the spatiotemporal tensor. After obtaining the spatiotemporal hidden layer features, it reverses the output of each hidden layer to obtain a predicted spatiotemporal feature $\bar{H}^t$. Subsequently, the center point prediction module utilizes a convolutional layer to decode this predicted spatiotemporal feature into the predicted center heatmap P^t with dimensions of $1 \times H \times W$ for the current frame.

$$\bar{H}^t = ConvLSTM(H^t) \tag{2}$$

$$P^t = Conv(\bar{H}^t) \tag{3}$$

Then, the Hadamard product is utilized to aggregate the predicted center heatmap P^t and deep semantic representations for obtaining fusion representations $\bar{z}^t$.

$$\bar{z}^t = z^t \circ P^t \tag{4}$$

Such a method can strengthen the features in the semantic representation that are related to the object's position, while suppressing background features unrelated to the object, thereby enhancing the performance of animal detection. Finally, we utilize the detection network to generate detection boxes and appearance feature vectors for each animal in the current frame, which are used for subsequent multi-object trajectory matching.

3.3. Object and trajectory calibration module

For an animal a_i in the t-th video frame, MRT-IB designates the position of the animal a_i as the center coordinates of the detection box [x_i, y_i]. The appearance of the animal a_i consists of two parts: first, the size of the animal, represented by the width and height of the detection box [w_i, h_i]. The second part is the appearance feature vector e_i. In MRT-IB, the detection box and the feature vector are used as information to distinguish between different animals. The trajectory quintuple τ_i for a_i is defined as follows:

$$\tau_i = [x_i, y_i, w_i, h_i, e_i] \tag{5}$$

where each value in the trajectory τ_i represents the detection box information and feature vector of the matched object for animal a_i in the $t-1$ frame.

Due to the inherent continuity in the movement trajectories and size changes of animals, MRT-IB considers using the Kalman filter algorithm to obtain the best estimate of the animal's detection box when matching with the trajectory. Taking the horizontal coordinate x_i^t of animal a_i at time step t as an example, let the best estimate of the horizontal coordinate after Kalman filtering for animal a_i and trajectory j be denoted as x_{ij}. In consecutive frames, an animal's movement trajectory can be considered as uniform linear motion. Generally, the greater the speed, the greater the distance of position change should be, and vice versa. If there is a certain contradiction between these two values, it indicates that the current observation may be affected by some noise. The discrepancy can be corrected using the relationship between position and velocity information to obtain the best estimated mean x_{ij}. The algorithm can first obtain the predicted value s^t of the animal in its current state, where t represents the current state, $\bar{x}_{ij}$ represents the predicted position, and $\bar{v}_{ij}$ represents the predicted velocity.

$$s^t = \begin{bmatrix} \bar{x}_{ij} \\ \bar{v}_{ij} \end{bmatrix} \tag{6}$$

Based on the relationship between velocity and displacement, under the condition that the video frame rate is sufficiently high, it can be approximated that each animal is undergoing uniform motion. Ignoring acceleration, the state transition equation for s^t can be derived as follows:

$$s^t = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} s^{t-1} = F s^{t-1} \tag{7}$$

The state transition matrix is denoted as F, and the relationship between velocity and displacement in any adjacent frames can be defined as mutually constraining R, represented by the covariance of position information x and velocity information v. Since the actual state transition of animals may be affected by external factors such as sudden movements or collisions, leading to certain variations in their state distribution, a noise distribution is introduced with its covariance Q^t represented as shown below.

$$R^t = F^t R^{t-1} (F^{t-1})^T + Q^t \tag{8}$$

$$R^t = \begin{bmatrix} \sum_{xx} & \sum_{xv} \\ \sum_{vx} & \sum_{vv} \end{bmatrix} \tag{9}$$

Simultaneously, based on the detection network, one can obtain the observed values of the current positions of the animals. These observed values are influenced by various factors such as the precision of the detection network itself, the posture of the animals during detection, and the background, and thus often have certain deviations. Let the mean of the observed value distribution be m^t and the variance be U^t. Meanwhile, since it is not possible to directly obtain the animals' velocities from the detection network, the Kalman filter introduces an observation matrix K^t, which allows the algorithm to

derive the predicted observation value $K^t s^t$ from the current state's predicted value s^t. To reconcile the predicted value $K^t s^t$ with the observed value m^t and have their distributions multiplied, resulting in a new distribution, the Kalman gain K is used, as shown below.

$$K = R^t R^t (R^t)^T (R^t R^t (H^t)^T + U^t)^{-1} \tag{10}$$

$$\begin{bmatrix} x_{ij} \\ v_{ij} \end{bmatrix} = s^k + H^t K (m^t - H^t s^t) \tag{11}$$

The final value of $\begin{bmatrix} x_{ij} \\ v_{ij} \end{bmatrix}$ is the best estimate of the match between object a_i and trajectory j after being processed by the Kalman filter. Other parameters can be derived in a similar manner to obtain an optimal estimated quadruple $[x_i, y_i, w_i, h_i]$. The loU (largest overlapping Union) algorithm is then used to calculate the similarity between the current trajectory and the detection box d_{ij}^b.

Furthermore, for any given animal a_i, an appearance feature vector e_i can be extracted. To represent the similarity between it and the most recently matched feature vector along the trajectory, the cosine metric can be employed as follows:

$$d_{ij}^e = \frac{e_i e_j}{\|e_i\| \, \|e_j\|} \tag{12}$$

After obtaining the similarity between animal a_i and trajectory j on the detection box, and the similarity between the appearance feature vectors, we have:

$$d_{ij} = w_b d_{ij}^b + w_e d_{ij}^e \tag{13}$$

where d_{ij} denotes the similarity between the i-th animal in the current frame and the j-th active trajectory in the current system.

3.4. Overall loss function

During the object extraction phase, a re-ID branch is introduced for extracting animal appearance features. After this branch extracts the feature vector of the animal, a fully connected network is used to obtain a confidence vector $P = \{p(c)|c \in [1, N]\}$, where N is the total number of ID labels used to distinguish different animals during training. For the animal a_i, there is a one-hot vector $V^i(c)$ used to indicate its assigned ID label. The cross-entropy loss function for classification constraint is as follows:

$$L_r = -\sum_c \sum_i V^i(c) \log(p(c)) \tag{14}$$

The center point prediction requires supervision during training to enable the module to infer the current frame's predicted heatmap P through the heatmap spatiotemporal tensor I. To this end, MRT-IB employs the structural similarity loss function (SSIM) as a constraint, which quantifies the similarity between two images by comparing their brightness, contrast, and structural information. The label heatmap is denoted as $\hat{P}$, with the brightness of the heatmap represented by its mean μ and the contrast represented by

its variance σ. The structural relationship between the predicted heatmap and the label heatmap is captured by their covariance $\Sigma_{p\hat{p}}$. The SSIM loss function L_p is given by:

$$L_p = \frac{(2\mu_p\mu_{\hat{p}} + c_1)(2\sigma_{p\hat{p}} + c_2)}{(\mu_p^2 + \mu_{\hat{p}}^2 + c_1)(\sigma_p^2 + \sigma_{\hat{p}}^2 + c_2)} \tag{15}$$

where μ_P and $\mu_{\hat{P}}$ are the mean values of the predicted and label heatmaps, respectively, $\Sigma_{P\hat{P}}$ is their covariance, and c_1 and c_2 are constants to prevent instability when the mean or covariance are near zero.

Furthermore, MRT-IB employs a detection box loss to guide the accurate detection of animals:

$$L_k = -\frac{1}{N}\left\{\begin{array}{cc}(1-\hat{p}_{xyc})\log(\hat{p}_{xyc}) & p_{xyc}=1\\ (1-\hat{p}_{xyc})\hat{p}_{xyc}\log(\hat{p}_{xyc}) & otherwise\end{array}\right\} \tag{16}$$

where $\hat{p}_{xyc}$ denotes represents the output of the detection box at position $[x, y]$ for the category c.

According to the aforementioned losses, the overall loss function of MRT-IB is defined:

$$L = L_r + \alpha L_p + \beta L_k \tag{17}$$

where α and β are trade-off parameters.

4. Experiments

4.1. Set up

Dataset: The performance of the MRT-IB in animal detection is verified based on two datasets. The first is the MOT (Multiple-object Tracking Challenge) pedestrian tracking dataset, which is one of the most authoritative datasets in the field of multi-object tracking. It is used to demonstrate the robustness of the algorithm. The second dataset consists of sheep in farm surveillance videos, which is used to explore the effectiveness of the algorithm in research on multi-object tracking of animals in farms. Unlike general tasks such as detection and classification, where the division of training and validation data often follows a 4:1 ratio in favor of the training set, or using five-fold cross-validation, the effectiveness of multi-object tracking models is typically validated by arranging the dataset in sequential order of video frames. The images from the first half of each video sequence are used for training, while the images from the latter half are used to validate the model's performance.

Metric: (1) Multiple Object Tracking Accuracy (MOTA): It is used to evaluate the tracking process over a period of time, accounting for detection errors and track switches. By defining the total number of objects at any time t as G_t, the number of false positives as N_t, the number of false negatives as P_t, and the number of track switches as I_t, the higher the MOTA, the more accurate the onject detection and the more stable the assignment. The MOTA is calculated using the following formula:

$$MOTA = 1 - \frac{\sum_t (N_t + P_t + I_t)}{\sum_t G_t} \tag{18}$$

(2) The IDF1 Score (IDF1): It is used to assess the correctness of track assignment over a period of time. By defining the correctly assigned detections in a period as T, false positives as N, and false negatives as M, the IDF1 is higher if the ID assignment of the detected objects is more accurate during the multi-object tracking process. The calculation formula for IDF1 is given by:

$$IDF1 = \frac{2T}{2T + M + N} \tag{19}$$

(3) ID Switch(IDs): It is used to evaluate the stability of tracks over a period of time, and the formula is as follows:

$$IDs = \sum_{t} I_t \tag{20}$$

Furthermore, Recall and Precision are also used to verify the effectiveness of MAR-IB.

Implementation details: MRT-IB is conducted in the Ubuntu 18.04 operating environment. The model is implemented using Python 3.7 and the PyTorch 3.6 framework, and both training and testing of the model are completed on CUDA 10.1 with an RTX 3080. In the experiment, MRT-IB utilizes the DLA-34 network, which has been pretrained on the COCO detection dataset, as the backbone for feature extraction. The initial learning rate during the training process is set to 10^{-3}, which is reduced to 10^{-4} when the training reaches the 20-th epoch, with a total of 30 epochs for training. Throughout the training process, images in the dataset undergo standard data augmentation procedures such as rotation, flipping, and color space transformations.

4.2. Analysis of Object and Trajectory Matching

In MRT-IB, to verify the effectiveness of the object and trajectory matching, three methods for calculating similarity between the object and trajectory are discussed, as shown in Table 2: (1) the first method directly calculates the IoU distance between the object detection box and the trajectory box recorded in the trajectory, (2) the second method predicts the position of the trajectory in the current frame using the Kalman filtering algorithm, and also filters the object position when matching the object and trajectory. Then, the oU distance is used for similarity calculation. (3) the third method calculates the cosine distance between the feature vector recorded in the trajectory and the feature vector of the object in the current frame for similarity calculation.

There are some observations obtained from the results in Table 2: (1) In the context of surveillance video in a breeding farm, the MRT-IB method can achieve satisfactory tracking results by simply using the Intersection over Union (IoU) distance to calculate the similarity between each object and the detection boxes of the matched objects in the previous frame. This is because in a breeding farm, the movements of the animals are relatively slow for the majority of the time. With a fast detection frame rate, the changes in the detection boxes of the same animal between adjacent frames are not significant, thus yielding fairly good tracking outcomes. However, when sheep flock together or move towards each other, the detection boxes of different sheep are in close proximity, which makes it easier for the IDs of different sheep to be swapped. Therefore, the ID swapping is noticeably higher in the IoU-only scheme compared to other methods. (2) MRT-IB can

associate objects based on the appearance feature information of different animals, thus effectively improving the IDF1 and IDs. However, the degree of improvement is not as significant as that of the Kalman filter model. This indicates that in the context of animal breeding, the motion information of animals plays a more substantial role in enhancing tracking accuracy compared to their appearance information. This is largely due to the fact that the appearance similarity among animals in breeding farms is not as high. Moreover, if an animal is occluded in one posture and then reappears in a different posture, its appearance information actually undergoes significant changes. Considering feature information in such cases could lead to misguidance. On the other hand, by introducing the re-ID branch to utilize appearance information during the tracking process, the detection model's ability to extract object features can be effectively enhanced, thereby improving the detection results of the objects. (3) The combination of the Kalman filter and IoU achieves the best matching because during the association process between each sheep's detection box and the detection boxes saved in the trajectory, it predicts the current detection box parameters for that trajectory based on the changes of past detection boxes. By combining the current frame's detection box of each sheep and filtering it, an optimal estimated value is obtained, effectively improving the accuracy of target-trajectory association. As a result, both IDF1 and IDs are significantly enhanced.

Table 2. Analysis of object and trajectory matching in MRT-IB in terms of the animal dataset

method	MOTA	IDF1	Recall	Precision	IDs
IoU	0.8020	0.8210	0.8770	0.9710	121
IoU+ Feature similarity	0.8240	0.8420	0.8770	0.9710	94
IoU+ Kalman filtering	0.8420	0.8770	0.8770	0.9710	98

4.3. Convergence Analysis of MRT-IB with respect to Animal Dataset

In the performance evaluation of the MRT-IB algorithm applied to an animal dataset, we conducted a thorough convergence analysis. This analysis aimed to explore the performance of the algorithm in key performance indicators such as Precision, Multiple Object Tracking Accuracy (MOTA), and Loss as the number of iterations increased.

From the Fig. 2, it can be observed that the algorithm's Precision rapidly improved in the initial stages, growing from around 0.4 to close to 0.9, demonstrating the algorithm's efficient capability in identifying animal targets. This swift enhancement indicates that the MRT-IB effectively distinguishes targets from the background when processing the animal dataset, thereby reducing false positives and false negatives.

Similarly, the MOTA metric also shows an upward trend, gradually increasing from a lower baseline to above 0.8, reflecting the algorithm's accuracy and stability in tracking multiple animal targets. The improvement in MOTA suggests that the MRT-IB algorithm can effectively maintain the tracking of targets, even under the complex and variable behaviors of animals.

The decreasing trend in the Loss value further confirms the improvement in algorithm performance. In the early stages of iteration, the Loss value significantly reduced, indi-

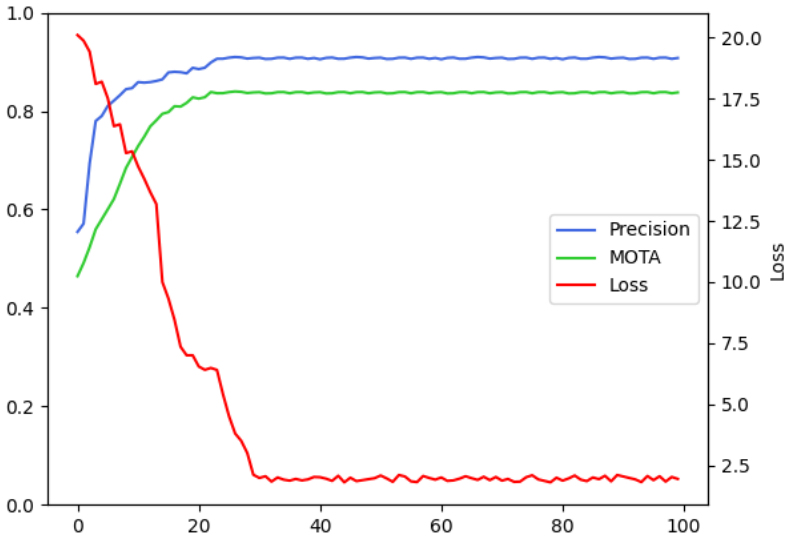

Fig. 2. Convergence Analysis of MRT-IB Performance on Animal Dataset

cating that the gap between the model's predictions and the actual annotations is quickly narrowing. As iterations progress, the Loss value tends to stabilize, suggesting that the model has converged to a more accurate predictive state.

Overall, the convergence analysis of MRT-IB on the animal dataset reveals its potential in multi-object tracking tasks. With an increase in the number of iterations, the algorithm shows significant improvement in key indicators such as Precision, MOTA, and Loss, ultimately reaching a stable high-performance state. These results indicate that MRT-IB is a promising algorithm suitable for animal behavior analysis and other scenarios requiring precise multi-object tracking.

4.4. Effectiveness of Center Point Prediction

The current multi-object tracking model, FairMOT [29], primarily considers the use of location and appearance information extracted by the detection network to associate targets with trajectories across consecutive frames. However, it overlooks the potential of utilizing this information during the detection phase to assist, thereby better aiding detection-based multi-object tracking models in completing tracking tasks. In previous experiments, it was observed that detection-based models could achieve fairly good results by relying solely on the position information, and the accuracy of the detection results could significantly enhance tracking precision. Therefore, this paper considers the introduction of a Central Point Prediction Module into the detection network to explore the impact of incorporating position-based multi-frame information during the detection phase on the results of multi-object tracking. This module can leverage the continuity of individual

animal trajectories in the video and the relative relationships between multiple animal trajectories to predict the positions of each animal in the current frame. By combining these predictions with the semantic features of animals extracted from the current frame image by the feature extraction network, it can assist in animal detection and tracking. The experimental results are shown in Table 3

From the comparative experiments, it can be seen that in MOT-16 and MOT-20, MRT-IB has improved MOTA by 1.0 and 1.3 points, respectively. However, IDS has only shown a slight increase compared to the baseline model. In contrast, in MOT-17, MRT-IB has enhanced MOTA by 1.9 points, while ID has decreased by approximately 25% compared to the baseline model. This indicates that in MOT-16 and MOT-20, although MRT-IB can improve the baseline model's MOTA, it is mainly due to the enhancement of the model's detection capabilities. This is evident from the recall and precision rates of the two datasets, where recall has increased while precision has slightly decreased. Through MRT-IB, the baseline model is better able to detect targets that were previously undetected, but it also introduces some false positives. In the case of MOT-17, MRT-IB not only improves the model's detection capabilities but also enhances its trajectory maintenance capabilities. Unlike the first two datasets, MRT-IB shows a decrease in recall, while the precision increases.

The results above indicate that MRT-IB can leverage the continuity of object positions to enhance the tracking performance of the model in two aspects. On one hand, when an object that has appeared before is missed by the original model in the current frame, MRT-IB will strengthen such targets, thereby reducing the model's false negatives. However, this may also lead to the detection of targets that should have disappeared in the current frame, resulting in false positives. Such false positives can cause the tracking model to assign new IDs to these targets, thus weakening the model's trajectory maintenance capability. On the other hand, when the original model produces false positives in the current frame, these targets, having not appeared before, will be relatively weakened, thereby improving the model's precision. But this could also lead to the weakening of new targets that should have been detected, which in turn reduces the recall rate of the tracking model. Meanwhile, as MRT-IB can improve detection precision, it can effectively enhance the tracking model's ability to maintain trajectories. Across the three datasets, MRT-IB has effectively increased the model's IDF1, indicating that MRT-IB can provide more precise information for the association phase, thereby helping the tracking model to improve the accuracy of matching trajectories to targets during tracking.

Table 3. Enhancement of the Central Point Prediction on the MOT Dataset for the Baseline Model FairMOT

method(dataset)	MOTA	IDF1	Recall	Precision	IDs
FairMOT(MOT-16)	0.6760	0.6940	0.7340	0.9370	456
MRT-IB(MOT-16)	0.6860	0.7050	0.7500	0.9310	479
FairMOT(MOT-17)	0.6930	0.7350	0.7800	0.9110	477
MRT-IB(MOT-17)	0.7140	0.7490	0.7680	0.9410	357
FairMOT(MOT-20)	0.7290	0.7210	0.7640	0.9610	2900
MRT-IB(MOT-20)	0.7424	0.7300	0.7830	0.9550	3014

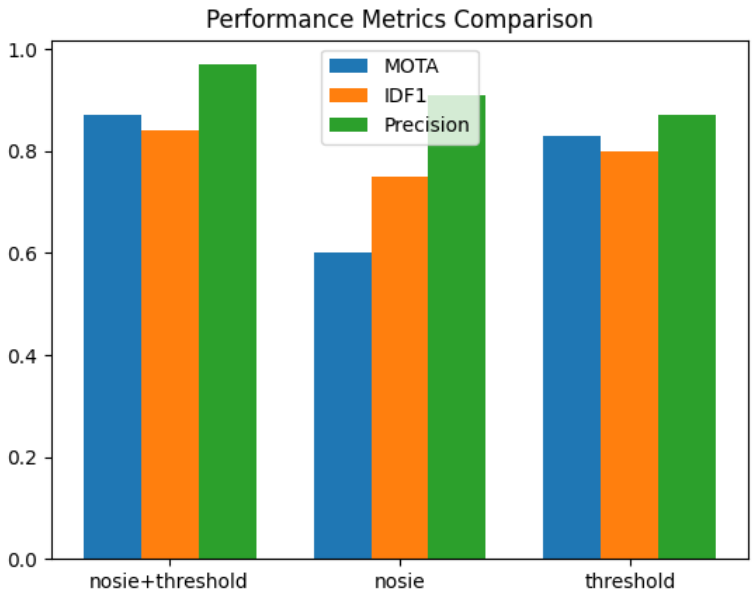

Fig. 3. Influence of noise and threshold on center point prediction

4.5. Influence of noise and threshold on Center Point Prediction

In real tracking scenarios, networks often encounter situations of missed detections and false alarms. However, during training, using real labels directly cannot simulate the missed detections and false alarms that occur in real scenarios. Therefore, it is necessary to process the input heatmaps during training to simulate missed detections and false alarms, thereby improving the robustness of the position prediction network. One intuitive method is to directly train with continuous sequences of images. When training the position prediction network, the heatmaps generated by the model in the previous few frames are used directly as the input to the position prediction network. Although this method allows the position prediction model to use the network's output during training, thereby improving robustness, it introduces fixed temporal continuity during training, which can lead to decreased generalization performance and overfitting. Additionally, in the early stages of training the detection network, the quality of the output of the detection network is not guaranteed, which can lead to the position prediction module learning irreversible noise. Therefore, during the training phase, for any frame image, this study extracts the true keypoint heatmaps of the previous frames and adds noise to these labels to simulate missed detections and false alarms. Specifically, this is achieved by randomly removing annotated points from the heatmap to simulate missed detections, while randomly adding new annotated points near the annotated points to simulate false alarms. This approach allows for the use of random image inputs during training to avoid the model being affected

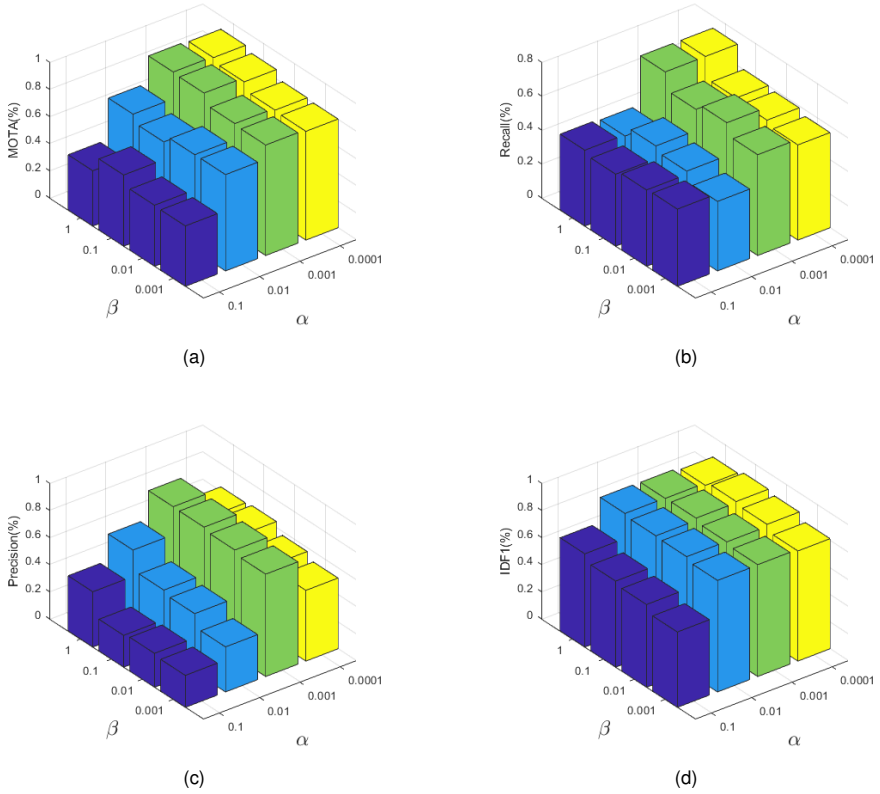

Fig. 4. The sensitivity analysis of parameters α and β on the animal dataset for MAR-IB

by fixed temporal data. During testing, the network produces small responses for many regions without targets. Although these responses do not appear in the results due to not reaching a predefined threshold, they can introduce considerable noise if the unprocessed keypoint heatmaps are directly fed into the central position module.

From the results in Fig 3, it can be observed that without threshold processing, multi-object tracking metrics experience a significant decline, while not adding noise to the labels also leads to a slight decrease in multi-object tracking metrics. To further analyze the experimental results at a deeper level, it is necessary to compare the various detection metrics with the trajectory matching metrics, as shown in Table 4.

Considering the responses in the heatmaps that do not reach a certain threshold during the testing process greatly reduces the accuracy of the tracking network, resulting in numerous false alarms. Consequently, many IDs are erroneously assigned, further weakening the network's ability to maintain trajectories. As a result, all metrics for multi-object tracking experience a significant decrease. However, after adding noise to the training labels, the tracking model effectively improves both recall and precision. This indicates

Table 4. Influence of noise and threshold on center point prediction in multi-object tracking scenario

method	noise+threshold	noise	threshold
Recall	0.7680	0.7660	0.7580
IDs	16.52	15.85	16.26

a notable enhancement in the detection capability of the tracking model, as well as an increase in precision in trajectory assignment and maintenance.

4.6. Parameter Analysis

Fig. 4 presents the outcomes of the sensitivity analysis for the trade-off parameters α and β within the MAR-IB framework when applied to the animal dataset. These parameters were confined to the values within sets $\{0.0001, 0.001, 0.01, 0.1\}$ and $\{0.001, 0.01, 0.1, 1\}$. The graphical representation in Fig. 4 highlights the resilience of MAR-IB to variations in α and β. Notably, the model exhibits consistent and reliable performance across different parameter settings, with an optimal balance achieved when α and β are both set to 0.001 and 1. This suggests that the model is relatively insensitive to fine-tuning of these parameters, which is advantageous for practical applications where computational resources may be limited or where the dataset's complexity necessitates a more straightforward tuning process.

5. Conclusion

In this paper, MRT-IB first highlights the limitations of current object detection algorithms in farm imagery, which are unable to automatically associate detection results of specific animals across consecutive frames for statistical analysis. It then introduces a detection-based multi-object tracking scheme that leverages the continuity of individual animals' movement trajectories and appearance changes in farm imagery, as well as the differences in position and appearance among different animals, to associate detection results. Furthermore, it proposes enhancing the detection capability in multi-object tracking by utilizing animal position information. Finally, this chapter designs multiple control experiments for the multi-object tracking model in farm imagery. The experimental results show that in the process of multi-object tracking of animals in farms, the animals' positions can effectively serve as a reference for data association. The animal appearance information extracted by the re-ID branch can effectively supplement the animal position information, improving the network's feature extraction ability and leading to more accurate detection results. Through ablation experiments on pedestrian datasets and animal tracking datasets, the chapter demonstrates that the model based on center point prediction proposed in this study can further effectively utilize object position information to enhance the model's detection capability and help the tracking model achieve more precise animal ID associations. The performance improvement of this module in multi-object tracking models is evident in multiple scenarios, proving the robustness of the model.

References

1. Ahrendt, P., Gregersen, T., Karstoft, H.: Development of a real-time computer vision system for tracking loose-housed pigs. Computers and Electronics in Agriculture 76(2), 169–174 (2011)
2. Alameer, A., Kyriazakis, I., Bacardit, J.: Automated recognition of postures and drinking behaviour for the detection of compromised health in pigs. Scientific reports 10(1), 13665 (2020)
3. Alwassel, H., Giancola, S., Ghanem, B.: Tsp: Temporally-sensitive pretraining of video encoders for localization tasks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3173–3183 (2021)
4. Bao, J., Xie, Q.: Artificial intelligence in animal farming: A systematic literature review. Journal of Cleaner Production 331, 129956 (2022)
5. Bergmann, P., Meinhardt, T., Leal-Taixe, L.: Tracking without bells and whistles. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 941–951 (2019)
6. Cernek, P., Bollig, N., Anklam, K., Döpfer, D.: Hot topic: Detecting digital dermatitis with computer vision. Journal of dairy science 103(10), 9110–9115 (2020)
7. Chen, C., Zhu, W., Steibel, J., Siegford, J., Wurtz, K., Han, J., Norton, T.: Recognition of aggressive episodes of pigs based on convolutional neural network and long short-term memory. Computers and Electronics in Agriculture 169, 105166 (2020)
8. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018)
9. Chen, L., Ai, H., Zhuang, Z., Shang, C.: Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In: 2018 IEEE international conference on multimedia and expo (ICME). pp. 1–6 (2018)
10. Gan, H., Ou, M., Zhao, F., Xu, C., Li, S., Chen, C., Xue, Y.: Automated piglet tracking using a single convolutional neural network. Biosystems Engineering 205, 48–63 (2021)
11. Gao, J., Liu, M., Li, P., Zhang, J., Chen, Z.: Deep multiview adaptive clustering with semantic invariance. IEEE Transactions on Neural Networks and Learning Systems (2023)
12. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
13. Kang, X., Zhang, X., Liu, G.: Accurate detection of lameness in dairy cattle with computer vision: A new and individualized detection strategy based on the analysis of the supporting phase. Journal of dairy science 103(11), 10628–10638 (2020)
14. Kolarevic, J., Aas-Hansen, Ø., Espmark, Å., Baeverfjord, G., Terjesen, B.F., Damsgård, B.: The use of acoustic acceleration transmitter tags for monitoring of atlantic salmon swimming activity in recirculating aquaculture systems (ras). Aquacultural engineering 72, 30–39 (2016)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM 60(6), 84–90 (2017)
16. Lee, S., Ahn, H., Seo, J., Chung, Y., Park, D., Pan, S.: Practical monitoring of undergrown pigs for iot-based large-scale smart farm. IEEE Access 7, 173796–173810 (2019)
17. Li, P., Gao, J., Zhang, J., Jin, S., Chen, Z.: Deep reinforcement clustering. IEEE Transactions on Multimedia (2022)
18. Matthews, S.G., Miller, A.L., PlÖtz, T., Kyriazakis, I.: Automated tracking to measure behavioural changes in pigs for health and welfare monitoring. Scientific reports 7(1), 17582 (2017)
19. Ren, S., He, K., Girshick, R., Sun, J., Faster, R.: Towards real-time object detection with region proposal networks, adv. Neural Inf. Process 28 (2015)
20. Su, Q., Tang, J., Zhai, M., He, D.: An intelligent method for dairy goat tracking based on siamese network. Computers and Electronics in Agriculture 193, 106636 (2022)
21. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. pp. 6105–6114 (2019)

22. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10781–10790 (2020)
23. Wang, X., Zhang, R., Kong, T., Li, L., Shen, C.: Solov2: Dynamic and fast instance segmentation. Advances in Neural information processing systems 33, 17721–17732 (2020)
24. Wang, Z., Zheng, L., Liu, Y., Li, Y., Wang, S.: Towards real-time multi-object tracking. In: European conference on computer vision. pp. 107–122 (2020)
25. Wutke, M., Heinrich, F., Das, P.P., Lange, A., Gentz, M., Traulsen, I., Warns, F.K., Schmitt, A.O., Gültas, M.: Detecting animal contacts—a deep learning-based pig detection and tracking approach for the quantification of social contacts. Sensors 21(22), 7512 (2021)
26. Yang, A., Huang, H., Zhu, X., Yang, X., Chen, P., Li, S., Xue, Y.: Automatic recognition of sow nursing behaviour using deep learning-based segmentation and spatial and temporal features. Biosystems Engineering 175, 133–145 (2018)
27. Van der Zande, L.E., Guzhva, O., Rodenburg, T.B.: Individual detection and tracking of group housed pigs in their home pen using computer vision. Frontiers in animal science 2, 669312 (2021)
28. Zhang, L., Gray, H., Ye, X., Collins, L., Allinson, N.: Automatic individual pig detection and tracking in pig farms. Sensors 19(5), 1188 (2019)
29. Zhang, Y., Wang, C., Wang, X., Zeng, W., Fairmot, W.L.: On the fairness of detection and re-identification in multiple object tracking. DOI: https://doi. org/10.1007/s11263-021-01513-4 pp. 3069–3087 (2021)

Acknowledgments. This work was supported by the: 1. Science and Technology Research Project of Henan Provincial Science and Technology Department in the 2021 year "Real-time cattle status Recognition and disease early Warning Application Research based on Convolutional Representation Flow" (Project number: 212102210138); 2. Annual science and Technology Research project of Henan Provincial Science and Technology Department in the 2023 year "Application Research of Pig status and Behavior Recognition and disease Warning based on space-time and Multi-objectives" (Project number: 232102210082); 3. Annual Science and Technology Research Project of Henan Provincial Science and Technology Department in the 2022 year "Application Research of Vision Transformer in Bovine Body Recognition Based on Adaptive Patch Segmentation" (Project number: 222102210280).

Fei Wang is with the School of Information Engineering, Henan University of Animal Husbandry Economy. Research direction: image processing, artificial intelligence.

Bin Xia is with the Zhengzhou Cotton&Jute Engineering Technology and Design Research Institute, All China Federation of Supply and Marketing Cooperatives. Research direction: big data processing, pattern recognition.

Liwu Pan is with the School of Information Engineering, Henan University of Animal Husbandry Economy. Research direction: image processing, artificial intelligence.

Received: April 18, 2024; Accepted: June 29, 2024.