Computer Science and Information Systems 11(3):1001–1016 DOI: 10.2298/CSIS130915061J

A Computer Remote Control System Based on Speech Recognition Technologies of Mobile Devices and Wireless Communication Technologies

Hae-Duck J. Jeong, Sang-Kug Ye, Jiyoung Lim, Ilsun You, and WooSeok Hyun

Department of Computer Software
Korean Bible University
Seoul, South Korea
hdjjeong@gmail.com, pajamasi726@hanmail.net
{jylim, isyou, wshyun}@bible.ac.kr

Abstract. This paper presents a computer remote control system using speech recognition technologies of mobile devices and wireless communication technologies for the blind and physically disabled population as assistive technology. These people experience difficulty and inconvenience using computers through a keyboard and/or mouse. The purpose of this system is to provide a way that the blind and physically disabled population can easily control many functions of a computer via speech. The configuration of the system consists of a mobile device such as a smartphone, a PC server, and a Google server that are connected to each other. Users can command a mobile device to do something via speech; such as writing emails, checking the weather forecast, or managing a schedule. These commands are then immediately executed. The proposed system also provides blind people with a function via TTS(Text To Speech) of the Google server if they want to receive contents of a document stored in a computer.

Keywords: speech recognition technology, mobile device, Android, wireless communication technique.

1. Introduction

Speech recognition technology, which is able to recognize human speech and change to text, or to perform a command, has emerged as the 'Next Big Thing' of the IT industry. Speech recognition is technology that uses desired equipment and a service which can be controlled through voice without using items such as a mouse or keyboard. It also appeared as part of ongoing research in progress in 1950s, but was not popularized until the mid-2000s, with low voice recognition. Presently, related speech recognition technologies, which have been previously used limitedly for special-purposes, have been rapidly evolving because of the proliferation of portable computing terminals such as smartphones interconnected with the expansion of the cloud infrastructure [8].

One of the most prominent examples of a mobile voice interface is *Siri*, the voice-activated personal assistant that comes built into the latest iPhone. But voice functionality is also built into Android, the Windows Phone platform, and most other mobile systems, as well as many applications. While these interfaces still have considerable limitations, we are inching closer to machine interfaces we can actually talk to [7].

This paper presents a computer remote control system using speech recognition technologies of mobile devices and wireless communication technologies for the blind and physically disabled population [5], [6], [13]. These people experience difficulty and inconvenience using computers through a keyboard and/or mouse. The purpose of this system is to provide a way the blind and physically disabled population can easily control many functions of a computer via speech. The configuration of the system consists of a mobile device such as a smartphone, a PC server, and a Google server that are connected to each other. Users command a mobile device to do something via speech such as directly controlling computers, writing emails and documents, calculating numbers, checking the weather forecast, or managing a schedule. These commands are then immediately executed. The proposed system also provides blind people with a function via TTS (Text To Speech) of the Google server when they want to receive contents of a document stored in a computer.

In Section 2, a few related works and technologies of the proposed remote computer control system are discussed. Section 3 describes comparison of speech recognition rates of current speech recognition systems. Section 4 presents how the proposed system using speech recognition technologies is designed and implemented, and finally the conclusions are described in Section 5.

2. Related Works and Technologies

Related works and technologies of the proposed computer remote control system using speech recognition technologies of mobile devices and wireless communication technologies are Android, and speech recognition algorithms as follows.

2.1. Android

Android is a Linux-based open mobile platform for mobile devices such as smartphones and tablet computers. It is composed of not only an operating system, but also middleware, user interface (UI), browser, and application. It also includes C/C++ libraries that are used in components of various Android systems [3]. Figure 1 shows that Android system architecture is divided into five hierarchical categories: applications, application framework, libraries, Android runtime, and Linux kernel [1], [2], [9]. The proposed application was designed and developed on Android.

2.2. Speech Recognition Algorithms

Google Speech Recognition. Google uses artificial intelligence algorithms to recognize spoken sentences, stores voice data anonymously for analysis purposes, and cross matches spoken data with written queries on the server. Key problems of computational power, data availability and managing large amounts of information are handled with ease using `android.speech.RecognizerIntent` package [1]. Client application starts up and prompts user to input using Google Speech Recognition. Input data is sent to the Google server for processing and text is returned to client. Input text is passed to the natural language processing (NLP) server for processing using HTTP (HperText Transfer

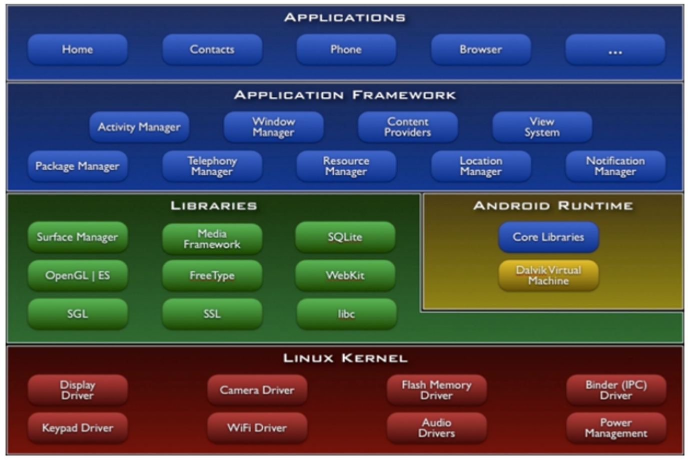


Fig. 1. Android system architecture.

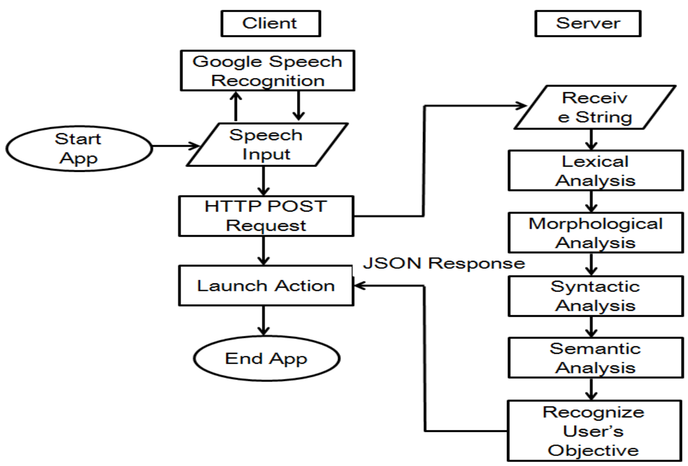


Fig. 2. Data flow diagram of speech recognition.

Protocol) POST [1]. Then the server performs NLP. Data flow diagram of speech recognition in Figure 2 shows that there are several steps involved in NLP as in the following:

1. **Lexical Analysis** converts sequence of characters into a sequence of tokens.
2. **Morphological Analysis** identifies, analyzes, and describes the structure of a given language's linguistic units.

[1] POST request is used to send data to a server. The string detected by speech recognizer is passed to the server using this method. It accomplishes this using in-built `HttpCore` API (i.e., `org.apache.http` package). The server performs processing and returns a JSON (JavaScript Object Notation) response. JSON is a lightweight data-interchange format, is based on a subset of the JavaScript programming language, and is completely language independent. In Java, `org.json.JSONObject` is used to parse strings [1].

3. **Syntactic Analysis** analyzes texts, which are made up of a sequence of tokens, to determine their grammatical structure.
4. **Semantic Analysis** relates syntactic structures from the levels of phrases and sentences to their language-independent meanings.

Hidden Markov Model. Modern general-purpose speech recognition systems are based on Hidden Markov Models (HMM). HMM is a doubly stochastic process with an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produce the sequence of observed symbols [4], [11]. HMMs are statistical models that output a sequence of symbols or quantities, and are used in speech recognition because a speech signal can be viewed as a piecewise stationary signal or a short-time stationary signal. In a short time-scales (e.g., 10 milliseconds), speech can be approximated as a stationary process. Speech can be thought of as a Markov model for many stochastic purposes [15]. Another reason why HMMs are popular is because they can be trained automatically and are simple and computationally feasible to use. In speech recognition, the hidden Markov model would output a sequence of n-dimensional real-valued vectors (with n being a small integer, such as 10), outputting one of these every 10 milliseconds. The vectors would consist of cepstral coefficients, which are obtained by taking a Fourier transform of a short time window of speech and decorrelating the spectrum using a cosine transform, then taking the first (most significant) coefficients. The hidden Markov model will tend to have in each state a statistical distribution that is a mixture of diagonal covariance Gaussians, which will give a likelihood for each observed vector. Each word, or (for more general speech recognition systems), each phoneme, will have a different output distribution; a hidden Markov model for a sequence of words or phonemes is made by concatenating the individual trained hidden Markov models for the separate words and phonemes.

The following notations for a discrete observation HMM are defined.
Let $T = \{1, 2, \cdots, T\}$ be the observation sequence (i.e., number of clock times), and T is length of the observation sequence. Let $Q = \{q_1, q_2, \cdots, q_N\}$ be states, where N is the number of states, $V = \{v_1, v_2, \cdots, v_M\}$ be discrete set of possible symbol observations, where M is the number of possible observations, $A = \{a_{ij}\}$ be state transition probability distribution, where $a_{ij} = Pr(q_i \text{ at } t+1 | q_i \text{ at } t)$, $B = \{b_j(k)\}$ be observation symbol probability distribution in state j, where $b_j(k) = Pr(v_k \text{ at } t | q_j \text{ at } t)$, and $\pi = \{\pi_i\}$ be initial state distribution, where $\pi_i = Pr(q_i \text{ at } t = 1)$ [11].

The mechanism of the HMM is explained in the following:

- Step-1. Choose an initial state, i_1, according to the initial state distribution, π.
- Step-2. Set $t = 1$.
- Step-3. Choose O_t, according to $b_{i_t}(k)$, the symbol probability distribution in state i_t.
- Step-4. Choose i_{t+1} according to $\{a_{i_t i_{t+1}}\}, i_{t+1} = 1, 2, \cdots, N$, the state transition probability distribution for state i_t.
- Step-5. Set $t = t + 1$; return to Step-3 if $t < T$; otherwise terminate the procedure.

We use the compact notation $\lambda = (A, B, \pi)$ to represent an HMM. For every fixed state sequence $I = i_1 i_2 \cdots i_\tau$, the probability of the observation sequence O is $Pr(O|I, \lambda)$, where

$$Pr(O|I, \lambda) = b_{i_1}(o_1) b_{i_2}(o_2) \cdots b_{i_\tau}(o_\tau). \tag{1}$$

In other words, the probability of such a state sequence I is

$$Pr(I|\lambda) = \pi_{i_1} a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{\tau-1} i_\tau}. \quad (2)$$

The joint probability of O and I is simply the product of the above two terms,

$$Pr(O, I|\lambda) = Pr(O|I, \lambda) Pr(I|\lambda). \quad (3)$$

Then the probability of O is obtained by summing this joint probability over all possible state sequences:

$$Pr(O|\lambda) = \sum_{all I} Pr(O|I, \lambda) Pr(I|\lambda) \quad (4)$$

$$= \sum_{i_1, i_2, \cdots, i_\tau} \pi_{i_1} b_{i_1}(o_1) a_{i_1 i_2} b_{i_2}(o_2) \cdots a_{i_{\tau-1} i_\tau} b_{i_\tau}(o_\tau). \quad (5)$$

Neural Networks. Neural networks emerged as an attractive acoustic modeling approach in automatic speech recognition (ASR) in the late 1980s. Since then, neural networks have been used in many aspects of speech recognition such as phoneme classification, isolated word recognition, and speaker adaptation [12], [15]. In contrast to HMMs, neural networks make no assumptions about feature statistical properties and have several qualities making them attractive recognition models for speech recognition. When used to estimate the probabilities of a speech feature segment, neural networks allow discriminative training in a natural and efficient manner. Few assumptions on the statistics of input features are made with neural networks. However, in spite of their effectiveness in classifying short-time units such as individual phones and isolated words, neural networks are rarely successful for continuous recognition tasks, largely because of their lack of ability to model temporal dependencies. Thus, one alternative approach is to use neural networks as a pre-processing e.g. feature transformation, dimensionality reduction, for the HMM based recognition.

Other Speech Recognition Systems. Modern speech recognition systems use various combinations of a number of standard techniques in order to improve results over the basic approach described above. A typical large-vocabulary system would need context dependency for the phonemes (so phonemes with different left and right context have different realizations as HMM states). It would use cepstral normalization to normalize for different speaker and recording conditions. For further speaker normalization it might use vocal tract length normalization (VTLN) for male-female normalization and maximum likelihood linear regression (MLLR) for more general speaker adaptation. The features would have so-called delta and delta-delta coefficients to capture speech dynamics and in addition might use heteroscedastic linear discriminant analysis (HLDA); or might skip the delta and delta-delta coefficients and use splicing and a linear discriminant analysis (LDA)-based projection followed perhaps by heteroscedastic linear discriminant analysis or a global semi-tied covariance transform (also known as maximum likelihood linear transform, or MLLT). Many systems use so-called discriminative training techniques that dispense with a purely statistical approach to HMM parameter estimation and instead optimize some classification-related measure of the training data. Examples are maximum mutual information (MMI), minimum classification error (MCE) and minimum phone error (MPE) [10], [15].

Fig. 3. Arirang note that is lyrical folk song in the Republic of Korea [14].

3. Comparison of Speech Recognition Rate

We have investigated how much recognition rates of current speech recognition systems, including Google speech recognition, NHN (Naver), Q Voice, S Voice, and Siri, are with Arirang [2], lyrical folk song in the Republic of Korea; and also see Arirang note in Figure 3.

One hundred replications in Korean were tested for each speech recognition system. According to our investigation, Table 1 shows that Google speech recognition system is the best of five speech recognition systems. Thus, it was used to design and implement our proposed system.

4. Implementation and Results

Figure 4 shows the architecture of the proposed system and command transmission methods among a mobile device, a Google server, and a personal computer server. The roles of each number are in the following:

1. A user commands using the speech recognition application of the mobile device.

[2] Arirang is a popular form of Korean folk song and the outcome of collective contributions made by ordinary Koreans throughout generations. Essentially a simple song, it consists of the refrain 'Arirang, arirang, arariyo' and two simple lines, which differ from region to region [14].

Table 1. Comparison of speech recognition rate for speech recognition systems.

Speech recognition system	Recognition rate (%)	Smartphone type	Smartphone version	Techniques used
Google speech recognition	100	Galaxy III	Android4.1.2	Google's own technology
NHN(Naver)	51	Galaxy III	Android4.1.2	Link
Q Voice	98	Optimus G	Android4.1.2	1st step: Google 2nd step: Wernicke
S Voice	96	Galaxy III	Android4.1.2	Vlingo
Siri	94	IPhone 5	IOS 6.1	Nuance

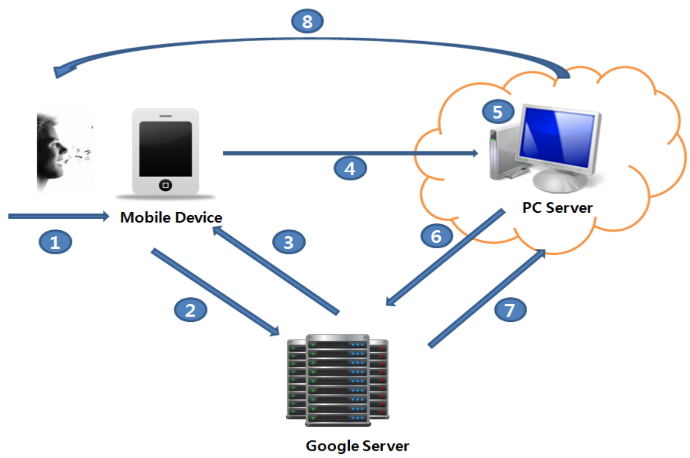


Fig. 4. Command transmission methods among a mobile device, a Google server, and a personal computer server.

2. Execute STT (speech to text) through the Google server.
3. Transmit results obtained from STT to the mobile device.
4. Transmit results obtained from STT to the personal computer server via wireless communications such as 3G, WIFI, and Bluetooth.
5. The personal computer server analyzes corresponding commands, and executes to distinguish between information which is sent to the Google server, and information which is executed on the personal computer server.
6. Transmit information to the Google server if there is information to use the Google server among commands.
7. The Google server returns corresponding values after analyzing corresponding services.
8. Give the user information received from the Google server with voice messages or execute.

Figure 5 shows overall use case diagram of the proposed system that contains more than five main functions such as speech recognition, keyboard control, mouse control, simple mode, and text transmission.

Fig. 5. Overall use case diagram of the proposed system.

Our proposed computer remote control system using speech recognition technologies of mobile devices and wireless communication technologies was implemented by Java programming language. The proposed application was designed and developed on Android as well.

4.1. Speech Recognition Mode

The below program code shows Java code of speech recognition for the proposed application. `startVoiceRecognitionActivity` fires an intent to start the speech recognition activity and `onActivityResult` handles the results from the recognition activity.

```
private void startVoiceRecognitionActivity() {
   Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
   intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
   RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
   intent.putExtra(RecognizerIntent.EXTRA_PROMPT, "Speech recognition demo");
   startActivityForResult(intent, VOICE_RECOGNITION_REQUEST_CODE);
 }
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
  if (requestCode == VOICE_RECOGNITION_REQUEST_CODE && resultCode == RESULT_OK) {
  // Fill the list view with the strings the recognizer thought it could have heard
  ArrayList<String> matches =
                       data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
  mList.setAdapter(new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1,
                                                                            matches));
  }
 super.onActivityResult(requestCode, resultCode, data);
}
```

Figure 6 shows speech recognition by touching the mobile device screen. When executing speech recognition by touching the top of the mobile device screen, all speech contents are typed and saved on the computer. When executing speech recognition by touching the bottom, corresponding service is executed by recognizing all speech contents. For example, a user commands the mobile device to do ’what is today’s weather?’ and then the remote system answers ’Today is 20 degrees Celsius and the weather is fine.’ Another example is that a user from the outside commands his/her mobile device to do ’Send meeting document in the document folder.’ and then the system finds it in the folder and transmits it to user’s mobile device or a personal computer that he/she wants.

4.2. Keyboard Control Mode

Figure 7 demonstrates computer keyboard control by touching the smartphone screen. A computer’s keyboard is controlled by a method that the key value entered by the user is transmitted from smartphone (client) to PC (server) through socket communication. The QWERTY keyboard, which is the most common modern-day keyboard layout, consists of XML. Each button has an independent `OnClickListener`, and depending on the state of the keyboard, transmitted values are different.

User-entered key values with the specified protocol (”$”) are sent to PC (server). The received values are stored on the PC (server) using `keypress ()` and `keyRelease ()` methods of the `Robot` class in Java.

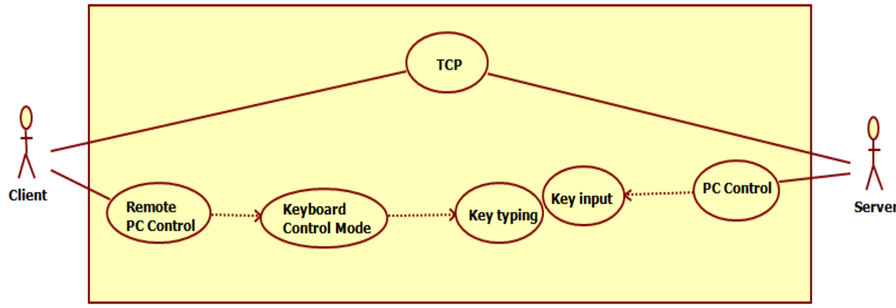
Fig. 6. Speech recognition by touching the smartphone screen.

4.3. Mouse Control Mode

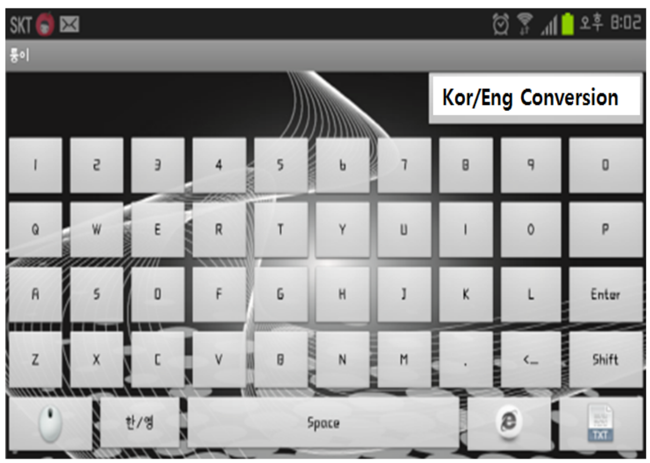
Figure 8 presents computer mouse control by touching the smartphone screen. There are double click, left click, and right click buttons. In order to control the mouse, using the touch screen of the smartphone (client), with UDP, the remote computer control system transmits the first coordinate and an actuated coordinate. In case of the mouse control, with UDP, speed rather than accuracy is prioritized because the system has to quickly transmit data. Using the `mouseMove ()` method of the `Robot` class in Java, the system remotely controls user's PC mouse pointer on PC (server) that was received the transmitted coordinates.

4.4. Simple Mode

Execution of applications users want on the simple mode is shown in Figure 9. While using a computer, there are programs that you often use, such as explorer, notepad, Hangul (Korean) word processor, GOM Player, and messenger. The Simple mode is the mode of execution that these programs are executed with a single click from a remote location. When the button is clicked on smartphone (client), the commands will be sent to the PC (server) through TCP communication. Using the `exec ()` method of the `Runtime` class in Java, with the touch of a button, the program that you want will be easily executed on PC received the commands through the external command.



(a) Use case diagram of keyboard control

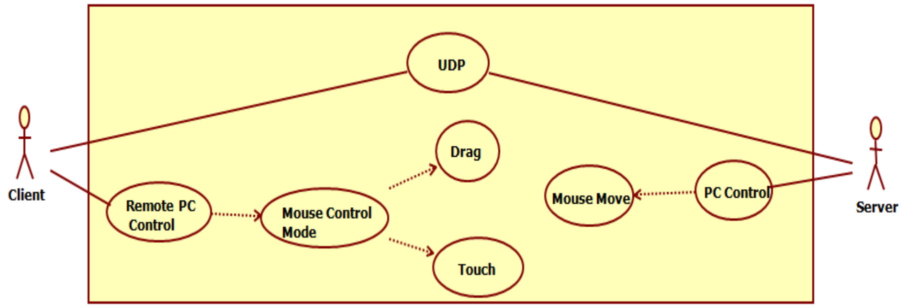


(b) Screenshot of keyboard control

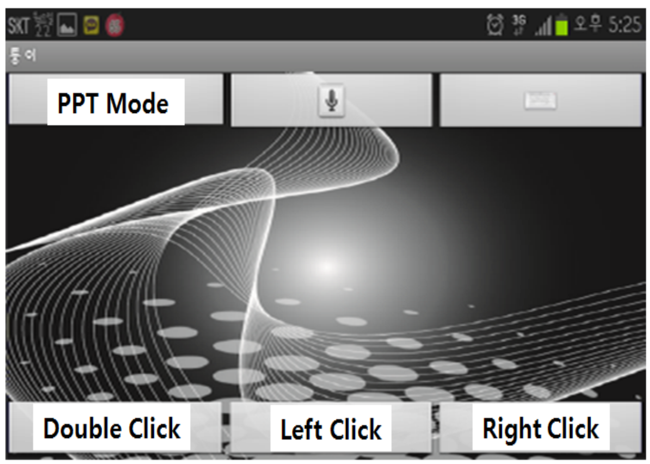
Fig. 7. Computer keyboard control by touching the smartphone screen.

4.5. Sending a Text Message Mode

The existing service method, which has transmitted texts through voice, does not read texts entered by the user and send back to the user. The proposed system, however, using the STT technology, provides the function that can correctly deliver the information since when the user inputs his/her voice on smartphone, it re-reads what you enter through the TTS function. When you have made all your input through `SmsManager`, the system sends a text message to the other party; and also see that Figure 10 shows a flowchart how to send a text message.



(a) Use case diagram of mouse control

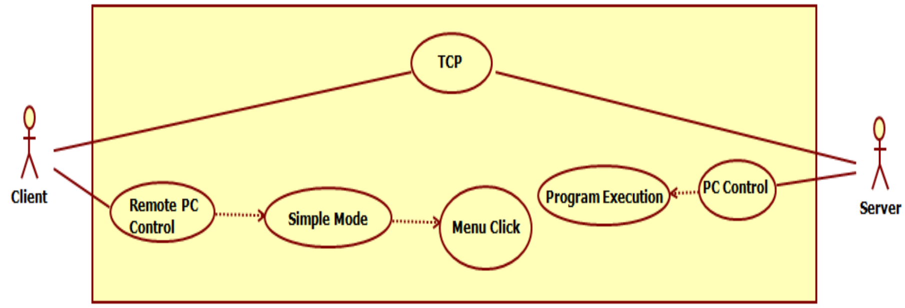


(b) Screenshot of mouse control

Fig. 8. Computer mouse control by touching the smartphone screen. There are double click, left click, and right click buttons.

4.6. Other Modes: File Search Mode

File search function is the ability to look at contents in the hard drive of the PC Server on smartphone. When smartphone users (Client) request a list of files in the PC (server), using the File class in Java, the proposed system distinguishes files and folders, and sends the list to the smartphone. This list with the folders and files shows on the smartphone screen through `ListView`. When the user clicks a folder, its contents shows in `ListView`. When the user clicks a file, the file is run through the `exec ()` method of `Robot` class on the PC Server. For example, when requesting to send `test.pdf` file from your smartphone, the `test.pdf` file, which is sent to your smartphone, can be found.



(a) Use case diagram of simple mode

(b) Screenshot of simple mode

Fig. 9. Execution of applications that users want to on the simple mode.

4.7. Other Modes: PC Screen Check Mode

When smartphone users request the transfer of your PC screen, the proposed system captures the current screen using the Robot class on PC, and transmits the screen to the smartphone through TCP communication. The smartphone receives the file and shows it on the `ImageView` screen. The multi-touch is possible, zooming in and out is feasible, and the system can check what the current PC's screen is. Commands with speech recognition are available, and a remote control mode in real time is possible by making sure the PC's screen.

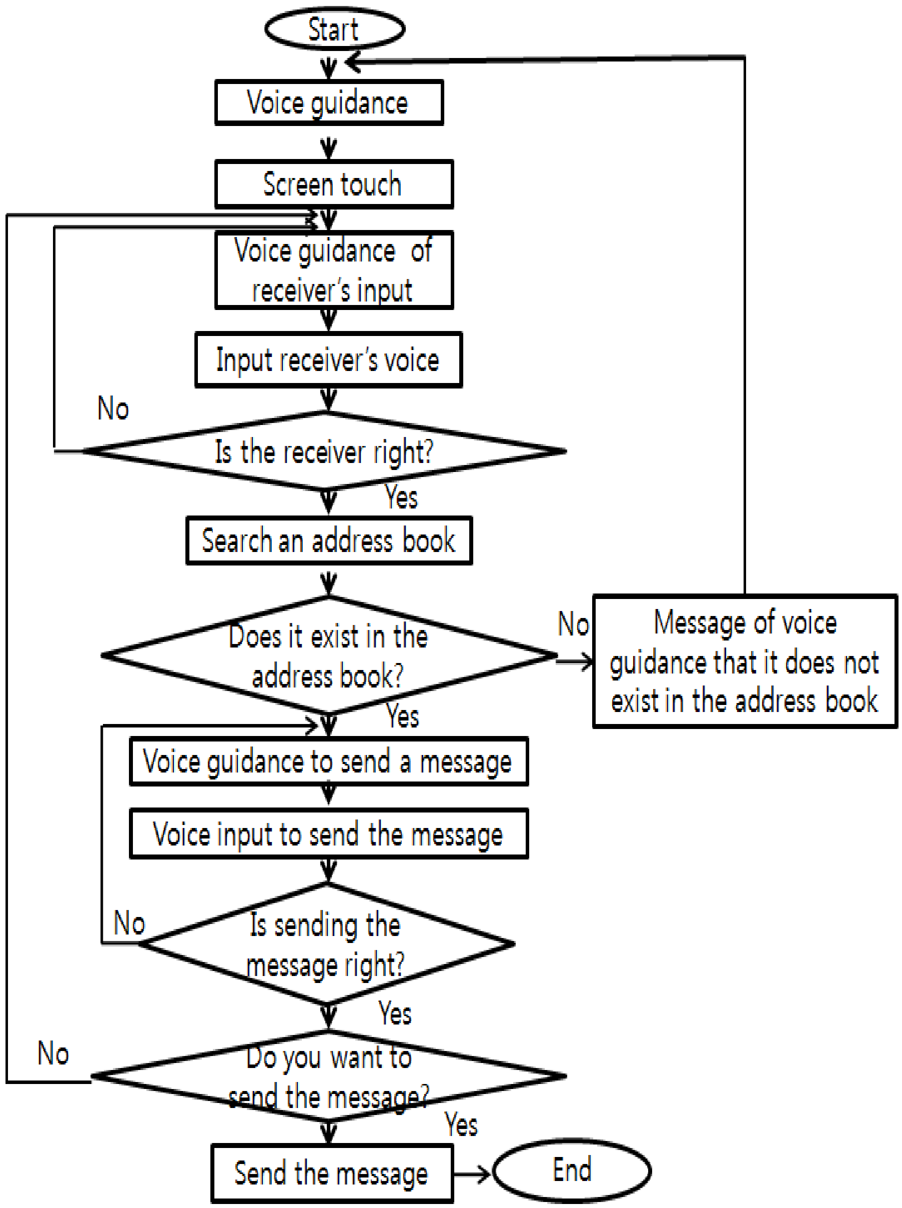


Fig. 10. Flowchart of sending a text message.

5. Conclusion

A computer remote control system using speech recognition technologies of mobile devices and wireless communication technologies for the blind and physically disabled population has been proposed. These people experience difficulty and inconvenience in using computers through a keyboard and/or mouse. The major purpose of this system was to provide a system so that the blind and physically disabled population can easily control many functions of a computer via speech. The system is very useful for the general population as well. Users command a mobile device to do something via speech such as directly controlling computers, writing emails and documents, calculating numbers, checking the weather forecast, or managing the schedule. These commands are then immediately executed. The proposed system also provides blind people with a function via TTS (text to speech) of the Google server if they want to receive contents of a document stored in a computer.

Acknowledgments. The authors would like to give thanks to the funding agencies for providing financial support. Parts of this work were supported by a research grant from Korean Bible University. The authors also thank Robert Hotchkiss and three referees for their constructive remarks and valuable comments.

References

1. Agarwal, A., Wardhan, K., Mehta, P.: JEEVES - A Natural Language Processing Application for Android. http://www.slideshare.net (2012)
2. Aguero, J., Rebollo, M., Carrascosa, C., Julian, V.: Does Android Dream with Intelligent Agents? Advances in Soft Computing 50, 194–204 (2009)
3. Android: Android Operating System, Wikipedia. http://en.wikipedia.org/wiki/Android OS
4. Jarng, S.S.: Analysis of HMM Voice Recognition Algorithm. Journal of Advanced Engineering and Technology 3(3), 241–249 (2010)
5. Jeong, H.D., Lim, J., Hyun, W., An, A.: A Real-time Location-based SNS Smartphone Application for the Disabled Population. Computer Science and Information Systems (ComSIS) 10(2), 747–765 (2013)
6. Jeong, H.D., Ye, S.K., Lim, J., You, I., Hyun, W., Song, H.K.: A Remote Computer Control System Using Speech Recognition Technologies of Mobile Devices. In: The Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing: Future Internet and Next Generation Networks (FINGNet-2013). pp. 595–600. Taichung, Taiwan (2013)
7. Knight, W.: Where Speech Recognition Is Going. MIT Technology Review, technologyreview.com (2012)
8. Korea Creative Contents Agency: Trends and Prospects of Speech Recognition Technologies (2011)
9. Lee, C.Y., An, B., Ahn, H.Y.: Android based Local SNS. Institute of Webcating, Internet Television and Telecommunication 10(6), 93–98 (2010)
10. Mao, Q.R., Zhan, Y.Z.: A Novel Hierarchical Speech Emotion Recognition Method Based on Improved DDAGSVM. Computer Science and Information Systems (ComSIS) 7(1), 211–221 (2010)
11. Rabiner, L., Juang, B.: An Introduction to Hidden Markov Models. IEEE ASSP Magazine pp. 4–16 (1986)
12. Tan, Z.H., Varga, I.: Network, Distributed and Embedded Speech Recognition: An Overview. Advances in Patterns Recognition (2008)
13. Torrente, J., Á.d. Blanco, Á. Serrano-Laguna, Vallejo-Pinto, J., Moreno-Ger, P., Fernández-Manjón, B.: Towards a Low Cost Adaptation of Educational Games for People with Disabilities. Computer Science and Information Systems (ComSIS) 11(1), 369–391 (2014)

14. UNESCO: Arirang, lyrical folk song in the Republic of Korea. http://www.unesco.org/culture/ich/RL/00445
15. Wikipedia: http://en.wikipedia.org/wiki/Speech_recognition

Hae-Duck Joshua Jeong is an associate professor in the Department of Computer Software at Korean Bible University, Seoul, South Korea. He received his Ph.D. in Computer Science and Software Engineering from the University of Canterbury, New Zealand. He is the author or co-author of more than 85 research publications, including more than twenty patents. Dr. Jeong is on the editorial board and a reviewer for various domestic and international journals. He is the corresponding guest editor or guest editor of ComSIS and MCM. His research interests include teletraffic modeling, stochastic simulation, multimedia telecommunication networks, intrusion detection system, social networking service, and real-time system. Member of IEEE NZ, KIPS, KSII, and ORSNZ.

Sang-Kug Ye is a student in the Department of Computer Software at Korean Bible University, Seoul, South Korea. He is the author or co-author of more than 5 research publications, including two patents. He received the President's Computer Software Competition Award four times including the grand prize twice from Korean Bible University from 2011 to 2013. His research interests include network security, intrusion detection system, and mobile applications.

Jiyoung Lim received her Ph.D. degree in Computer Science and Engineering from Ewha Womans University in 2001. She is currently an associate professor of Computer Software at Korean Bible University, Seoul, South Korea. Her research interests include wireless/sensor network security, and M2M network security.

Ilsun You received his Ph.D. degree in Computer Science from Dankook University, Seoul, South Korea in 2002. Also, he obtained his second Ph.D. degree from Kyushu University, Japan in 2012. In 2005, he joined Korean Bible University, South Korea as a full time lecturer, and he is now working as an associate professor. Dr. You is on the editorial board for various domestic and international journals. Also, he has served as a guest editor of several journals. His main research interests include Internet security, authentication, access control, formal security analysis, MIPv6, and ubiquitous computing.

Wooseok Hyun is the corresponding author of this paper. She is an associate professor in Computer Software at Korean Bible University, Seoul, South Korea. She received her Ph.D. in Computer Science from Gyeongsang National University, South Korea. She is the author or co-author of more than 30 research publications, including five patents; reviewer of various domestic and international journals. Her research interests include ubiquitous computing, intelligent system, fuzzy system, information retrieval system, and artificial intelligence. Member of KIISE, KIPS, KMMS.

Received: September 15, 2013; Accepted: January 21, 2014.