논문 2021-16-12

Development of a Work Management System Based on Speech and Speaker Recognition

Abdulaziz Gaybulayev, Jahongir Yunusov, Tae-Hyong Kim*

Abstract : Voice interface can not only make daily life more convenient through artificial intelligence speakers but also improve the working environment of the factory. This paper presents a voice-assisted work management system that supports both speech and speaker recognition. This system is able to provide machine control and authorized worker authentication by voice at the same time. We applied two speech recognition methods, Google's Speech application programming interface (API) service, and DeepSpeech speech-to-text engine. For worker identification, the SincNet architecture for speaker recognition was adopted. We implemented a prototype of the work management system that provides voice control with 26 commands and identifies 100 workers by voice. Worker identification using our model was almost perfect, and the command recognition accuracy was 97.0% in Google API after post- processing and 92.0% in our DeepSpeech model.

Keywords : Work management system, Speech recognition, Speaker identification, Smart factory, Voice assistance

Ⅰ. Introduction

Deep learning has brought revolutionary performance gains in numerous applications such as computer vision, natural language processing, and time-series data prediction. Deep learning is becoming more and more popular in the industry as it enables a higher level of automation and accuracy for most tasks and is now replacing existing technologies. Smart manufacturing is a new manufacturing paradigm in which machines are connected to a network, monitored by various sensors, and controlled by intelligent algorithms using deep learning techniques.

In smart manufacturing, machine motion is usually controlled and managed by dedicated software that provides machine motion monitoring, job history management, statistical information collection and visualization, and so on. If a machine is not fully automatically controlled, workers have to manually intervene in the operation of the machine. Mechanical buttons, joysticks, or soft buttons in management software are currently common tools for manual machine control.

Voice assistance, which has greatly improved its performance through deep learning, is now a useful and popular feature in real life. Voice recognition, such as Apple's Siri or Samsung's Bixby, is widely used to control smart devices, home appliances, and cars. However, the use of voice interface in manufacturing is still in its early stage because very high accuracy is required to control machines to prevent malfunctions or accidents. Nevertheless, voice has great potential for human-machine interfaces, as it can free the worker's hands.

Voice control is very useful when workers need to press soft buttons on the display with dirty fingers or work gloves on. It also helps when workers need to quickly stop machine operation in an emergency situation. Currently several solutions are emerging that add voice support to existing manufacturing systems.

Speaker recognition is the process of identifying or verifying a speaker using the speaker's voice. Speaker identification determines the identity of an unknown speaker, whereas speaker verification tests if a speaker's voice matches a particular speaker. Speaker recognition has been applied in phone-based customer service, criminal investigations, biometric banking, and so on. Worker authentication is necessary to verify approved workers for factory security. Employee identification (ID) cards with radio frequency ID (RFID) chips are commonly used, but they are not secure as they can be lost or stolen. Biometric authentication using fingerprints or voiceprints can be more secure if its accuracy is

*Corresponding Author (taehyong@kumoh.ac.kr)
Received: Mar. 16, 2021, Accepted: Apr. 6, 2021.
A. Gaybulayev: Kumoh National Institute of Technology (Ph.D. Student)
J. Yunusov: Kumoh National Institute of Technology (M.E.)
T.-H. Kim: Kumoh National Institute of Technology (Prof.)
※ This paper was supported by Research Fund, Kumoh National Institute of Technology (2018-104-079).

ⓒ IEMEK J. Embed. Sys. Appl. 2021 Jun. 16(3) 89-97
ISSN : 1975-5066
http://dx.doi.org/10.14372/IEMEK.2021.16.3.89

reliably high. As voice is used for speech recognition, speaker recognition can be performed additionally at the same time.

In this study, we designed a voice-assisted work management system (WMS) that uses both speech and speaker recognition. Integrating voice support into work management can provide a variety of useful features, including machine control by voice command, authorized worker authentication, and automatic worker-based job history management. We implemented a machine control system with two speech recognition models, online recognition using Google Cloud speech-to-text (STT) service [1] and offline recognition using a deep learning-based speech recognition engine. For worker recognition, a deep learning-based speaker recognition engine was adopted. We applied this system to a real manufacturing machine with 26 operation commands, and evaluated the performance of the implemented speech and speaker recognition in 100 worker situations.

The paper is organized as follows. First, we briefly review the existing research work that introduced voice assistance for industrial uses in section II. The proposed work management system based on speech and speaker recognition is explained in section III. The implementation of the system for a real manufacturing machine is described in detail in section IV. The performance results and evaluation of the implementation system are presented in section V with some discussion. Finally, section VI concludes our work with some final remarks.

II. Related Work

Speech recognition has a long history; it has been studied since about 1950s. From 1990s, speech recognition showed possibilities of practical uses with hidden Markov models (HMM) and feedforward artificial neural networks [2] and there were initial trials to introduce speech recognition in specific application domains such as healthcare [3] and robots [4, 5]. In 2012. Adam Rogowski examined the specific requirements that should be fulfilled by industrially oriented voice control systems for use in robotized manufacturing [6]. He proposed a special format for quasi-natural sublanguage syntax definition and a novel algorithm for semantic analysis with specific features of voice commands for controlling industrial devices and machines.

Recently, there was a study on speech recognition control of cameras for a surgical robot [7]. The authors used 7 commands to control cameras: 'left', 'right', 'down', 'up', 'forward', 'backward', and 'stop'. They proposed a new intentional speech control to control movement over long distances and voice-to-motion calibration to decrease the ambiguity of control. [8] and [9] presented voice-controlled human-machine interface systems for wheelchairs based on Arduino and Raspberry Pi. They also used 5 voice commands to control wheelchair movements.

Voice control can be used with other modes of user interfaces. Oliver Ohneiser and et al. proposed a controller working position prototype, TriControl, for air traffic control by integrating speech recognition, eye-tracking, and multi-touch sensing [10]. Wojciech Kaczmarek and et al. presented a control module for industrial robots by means of gesture and voice commands [11].

Since deep learning based speech recognition showed outstanding performances, it has been replacing traditional techniques. There also appeared several cloud services that provide application programming interfaces (APIs) for powerful online voice recognition using deep learning such as Google cloud speech API and Microsoft Azure speech-to-text API [12]. In [13] and [14], the authors presented a control and management system for home lighting that supports not only voice-control but also dialog service like chat-bot. They used both the Google cloud speech API and their own deep learning based intent recognition engine with a dialog control algorithm.

Speaker recognition as a method of authentication is now emerging in industrial applications as there is a need for security in voice-control systems. C. Shayamunda and et al. proposed a biometric authentication system for industrial applications using speaker recognition [15]. They used an autocorrelation voice activity detector and Gaussian mixture models to identify users of the system. However, the performance of their speaker recognition, error rate of 12.1% and verification time of 4.7 seconds, appears not ready for industrial application yet. Meanwhile, Eric Ke Yang and et al. examined voice cloning attack technology for construction of specific voice recognition system in 5G-aided industrial IoT domain [16] as voice cloning may lead to industrial accidents an other potential security risks.

III. The Proposed Approach

The proposed work management system aims to intelligently manage the work history by controlling machine operation and authenticating the worker at the same time with a voice interface. The system's voice

assistance can potentially increase work efficiency through quick control and reduce the risk of injury in emergency situations. The proposed system tries to apply the latest deep learning based speech and speaker recognition technology for the best performance.

1. Overview

The proposed system supports a cloud-based online speech recognition API or a deep learning-based offline speech recognition selectively, depending on the situation. Online speech recognition engines can usually recognize a vast vocabulary including all common words, with great accuracy. For industrial terms that are used less frequently, however, the recognition accuracy may be relatively low, which may require post-processing for the recognized words to determine the final control command. In addition, online API services require an Internet connection and are subject to charges based on usage, which can be a burden. On the other hand, using your own speech recognition engine requires a large amount of speech data for deep learning. As you know, more data is needed to increase accuracy. We used Google cloud speech API and DeepSpeech [17] respectively for online and offline speech recognition.

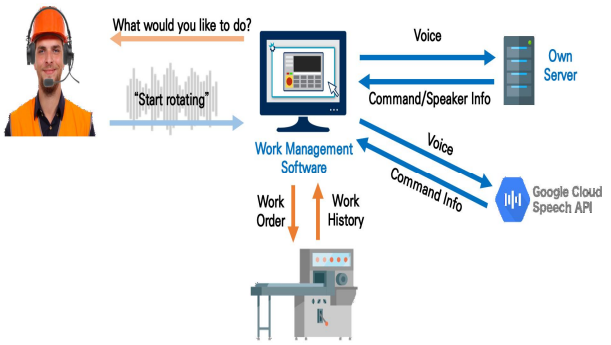

Fig.1. The proposed work management system

There are two categories of speaker recognition: text-dependent and text- independent. If the text is the same for registration and verification, speaker recognition is text-dependent. Otherwise, it is text-independent. A large amount of audio data for training is required to capture the unique phonetic characteristics of every speaker. If a small number of voice commands are used for machine control and these voices are used for speaker recognition, the text-dependent approach can be practical because it is simpler and more accurate. On the other hand, if the number of voice commands is rather large, the text-dependent approach can be cumbersome because voice registration is time-consuming. The proposed approach chose text-dependent speaker recognition, assuming that the number of voice commands is small enough to register. We used SincNet [18], a deep learning based speaker recognition model, which has a simple architecture but great performance.

An overview of the proposed work management system is illustrated in Fig.1.

2. Speech recognition

For offline speech recognition with our own server, DeepSpeech is used which is an open-source STT engine based on Baidu's deep learning based speech recognition model [19]. In 2014, Baidu Research presented the original Deep Speech whose architecture was significantly simpler than traditional speech systems but outperformed the existing state-of-the-art speech recognition systems at that time such as Google API and wit.ai [20]. Deep Speech needs neither hand-designed sound effect modeling components nor a phoneme dictionary. It directly learns such information by a well-optimized recurrent neural network (RNN) training system and a set of novel data synthesis systems. Fig. 2 depicts the structure of Deep Speech's RNN model.

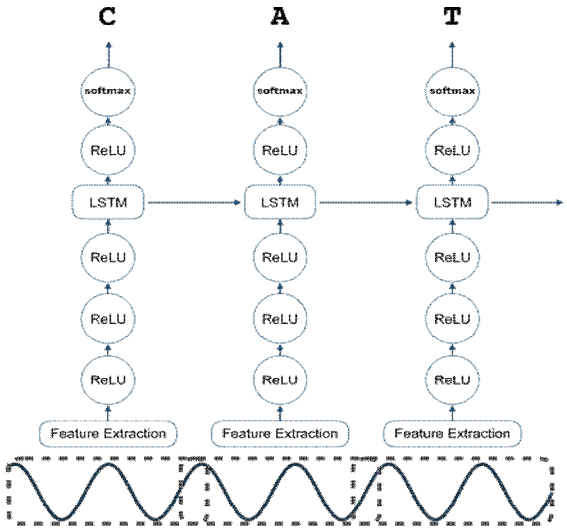

Fig.2 The RNN model of Deep Speech [21]

The RNN model of Deep Speech has 5 layers of hidden units among which only 4th layer is a recurrent layer. At each time step, the power at different frequencies is extracted by a sliding window and fed into three fully connected layers for capturing features. The next bidirectional RNN layer examines the condition of speech and the final layer estimates the probability of each character at each time step. The output probability

of the RNN model is passed to the connectionist temporal classification (CTC) layer to finally estimate the word sequence for the given utterance. Deep Speech also uses an N-gram language model and the word count to find out the next most likely word. The following objective function is used to find the optimal character sequence c from an utterance x, where α and β are hyperparameters and Plm(c) is the probability of the sequence c according to the N-gram model.

$$Q(c) = \log(P(c \mid x)) + \alpha \log(P_{lm}(c)) + \beta \, \mathrm{wordcount}(c).$$

3. Speaker recognition

SincNet, a novel convolutional neural network (CNN) architecture for speaker recognition, is applied in the proposed system. The CNN architecture of SincNet learns low-level speech representations from raw speech samples directly without using standard hand-crafted features. Fig.3 shows the architecture of SincNet for speaker classification.

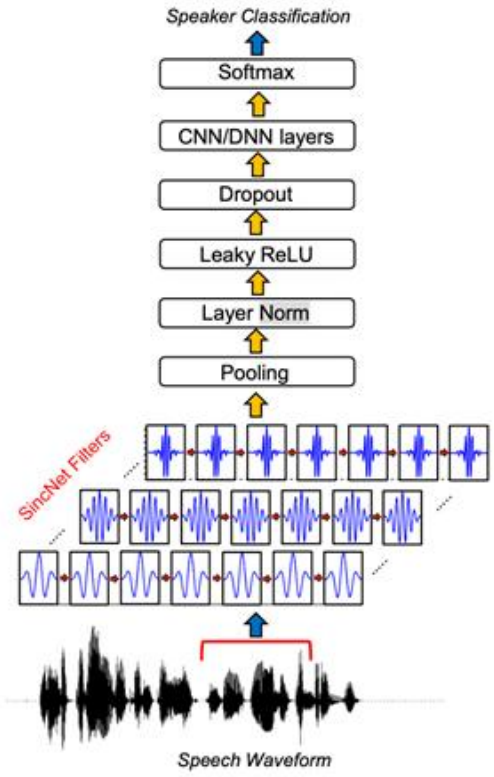

Fig.3. The architecture of SincNet [18]

The novel point of SincNet is the convolution filters designed to capture the characteristics of voice in different frequency bands. In order to make each convolution filter operate as a band-pass filter, SincNet applied a sinc function in convolution filters. The convolution operation of a SincNet filter can be represented as $y[n] = x[n] * g[n, f_1, f_2]$, where $x[n]$ is a chunk of the speech signal, $y[n]$ is the filtered output, and $g[n, f_1, f_2]$ is a filter-bank with two parameters f_1 and f_2 which are low and high cutoff frequencies for band passing respectively. The filter-bank function g is as follows:

$$g[n, f_1, f_2] = 2f_2 \,\mathrm{sinc}(2\pi f_2 n) - 2f_1 \,\mathrm{sinc}(2\pi f_1 n).$$

where $\mathrm{sinc}(x) = \sin(x)/x$. The sinc function in the time domain corresponds to the rectangular function in the frequency domain that is used to construct a band-pass filter.

SincNet filters can capture the formant frequencies much better than normal CNN filters which are related to acoustic responses of the human vocal tract. Accordingly, SincNet outperforms the existing CNN based speaker identification systems with various data sets.

Ⅳ. Implementation

We implemented the proposed work management system for an inspection machine of metal parts. Detailed information on the system components, data preparation, and training procedures for speech and speaker recognition is described in this section.

1. Construction

The implementation structure of the proposed work management system is shown in Fig.4.

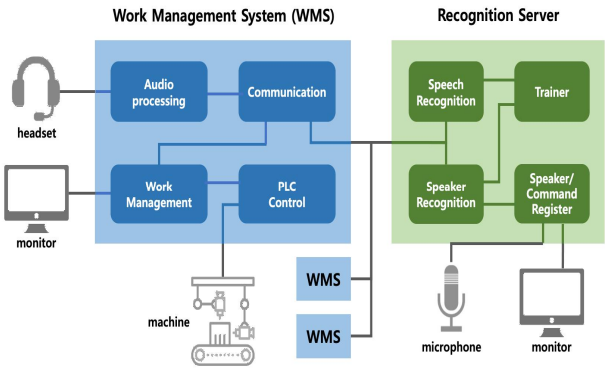

Fig.4. The implementation structure of WMS

The work management system is composed of an audio processing part for input voices from the connected headset, the communication part for delivering data and information, the PLC control part for controlling the connected machine, and the main work management part.

Workers should wear wireless headsets for clearer voices to be captured in noisy environments without being disturbed. For digital audio processing, we used a single channel, 16bit integer resolution, and 16kHz sampling rate. Since we developed the work management software in the C# programming language on the Microsoft Windows 10 operating system, NAudio, an open-source .NET audio library was used for handling audio data [22].

The digital audio stream is transmitted to the recognition server by the communication part. During the login process, both the speech and speaker recognition modules are the targets of the transmission. After logging in, the audio stream is transmitted only to the speech recognition module for simplicity. The recognition results from the recognition server are delivered to the main work management part by the communication part.

The work management part manages the entire working process of the worker, including worker login, display of recognition results, machine control with PLC drivers, work record storage and retrieval, statistical information of work records, and graphical user interface. As described before, it was developed in the C# programming language. Fig.5 shows a screenshot of the work management software.

Fig.5. A screenshot of the work management software

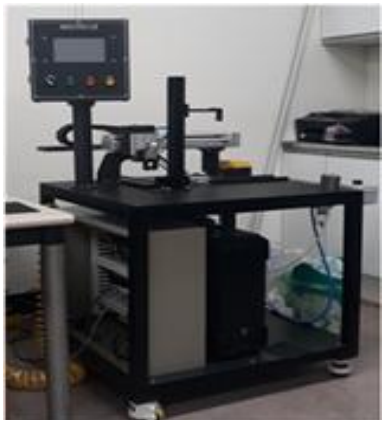

Fig.6. A prototype of the whole WMS

The work management system was packaged in a PC and attached to the machine for control as shown in Fig.6. The recognition server is connected to and serves multiple work management systems over the Internet. It has speech and speaker recognition parts, a speaker and command registration part, and trainer parts for additional training of speech and speaker recognition engines. Both the speech and speaker recognition parts segment the online streaming audio data to send appropriate inputs repeatedly to the speech and/or speaker recognition engines.

The speech recognition engine is based on DeepSpeech version 0.91 with Tensorflow 1.15 framework and the speaker recognition engine is based on a SincNet implementation with Keras functional APIs and Tensorflow 1.15. The training data preparation and training processes are explained in the subsequent subsections.

If there is a new worker which can operate a target machine, this worker has to be registered with his/her voice using the registration software of the recognition server for speaker recognition. The new worker must read a number of specific sentences presented in the connected display. When a new command is required to control machines, additional voice data for that command needs to be also recorded. If a new worker is registered or a new command for machine control is added, additional training has to be performed by the trainer module using transfer learning.

2. Data preparation for training

For training the deep learning models for speech and speaker recognition, we gathered and preprocessed audio files of voice commands. For voice control of the target machine, 26 short control commands were selected such as “forward rotation”, “backward rotation”, and “return to origin”. Note that all commands are Korean, and the maximum number of syllables for commands is only four. Some commands are paired, with only one or two syllables different. Most of the control commands in Korean are based on Chinese characters, which are not used frequently in everyday life. The peculiarity of these Korean control commands makes speech recognition difficult, even in the Google Speech API. That is why we had to perform post-processing to correct the commands recognized by the Google Speech API.

We gathered 100 people and collected 5 voices for each of the 26 commands. Additionally, In addition, we gave each person an 8-digit unique number as a worker

ID and recorded 5 voices of the worker ID for each person. This voice data, about 6 hours long, were used to train and test both the speech and speaker recognition networks. Actually, a large amount of voice data is required to obtain a commercial-level of recognition accuracy for both speech and speaker recognition. So we generated additional audio data for training with some augmentation techniques.

We designed an audio data augmentation code using librosa library [23] for changing audio speed and pitch, adding percussive effects, and so on. Using this code, we generated 107 hours of audio data for training, 12 hours of data for validation, and 8 hours of data for testing for 26 commands and worker IDs. DeepSpeech also provides various on-the-fly audio data augmentation techniques that can be used during training. We applied the following augmentation techniques during DeepSpeech training: overlaying noise data, adding reverberation, changing volume levels, putting frequency and time masks, and dropping out random data points.

For speaker augmentation, many of those augmentation techniques cannot be used because they may change the characteristics of voice which are required for speaker recognition. Therefore, we developed a special augmentation environment that can keep the voice characteristics as shown in Fig.7. In fact, this configuration is also an automatic testing environment for speech or speaker recognition that emulates live voice streaming with a microphone.

Instead of a real speaker's voice, the tester plays recorded voice files with a loudspeaker for automatic continuous testing. The target system for testing captures the played voice with a microphone and returns the recognition results to the tester. At that time, it also sends the captured audio data. If this testing environment is at usual living environments having everyday noise, those captured audio files can be used as augmented data. By changing the loudspeaker or the microphone, or their volume levels, other types of augmentation data can be obtained just for speaker recognition. Using this augmentation technique as well as some existing techniques, we finally generated 114 hours of audio data for training, 17 hours of data for validation, and 11 hours of data for testing.

3. Model Configuration and Training

For training a DeepSpeech network to recognize the machine control commands and worker IDs in Korean, we built a new language model and optimized the values of the model's hyperparameters for our data set. If 26 machine control commands are only the target for speech recognition, the language model is not required because each of 26 machine control commands can be regarded as a word. However, as a worker ID is a combination of digits, it cannot be considered a word. Accordingly, each digit in a worker ID has to be recognized separated as a word, which requires a language model. We generated a 3-gram trie language model using kenlm, an open-source language model inference code by Kenneth Heafield [24]. We applied this language model to the output layer of the DeepSpeech network and trained it to get the optimal values of hyperparameters α and β. Then we trained the network again with those optimized hyperparameters to get a sufficient recognition performance.

When training the SincNet model to identify 100 speakers, we changed the number of units of the last three fully connected layers from 2048 to 1024 in order to reduce computation cost. With this modified model, we followed the training procedure provided by the SincNet authors.

V. Evaluation

The implemented work management system is evaluated based on its performance of speech and speaker recognition. Basic performance metrics are the recognition accuracy and the recognition time. The detailed information of the data set for training, validation, and testing speech and speaker recognition models is shown in Table 1.

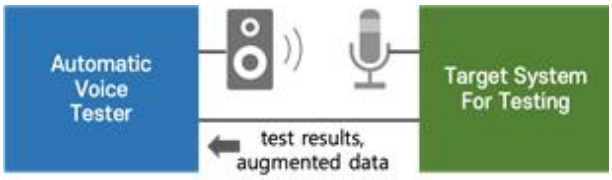

Fig.7. Data augmentation environment for speaker recognition

Table 1. Information of the data set

		train	val	test
speech recognition	no. of speakers	85	9	6
	no. of voices	172,960	18,000	12,176
speaker recognition	no. of speakers	100	100	100
	no. of voices	43,200	5,940	4,860

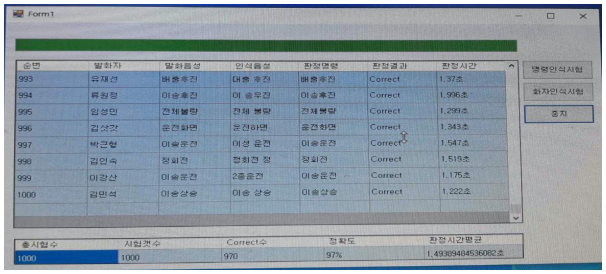

Fig.8. A screenshot of the testing software

We used the automatic testing environment for performance evaluation as shown in Fig. 7. In that environment, a testing software was set up in the tester that plays a random test audio file and retrieves the results from the implemented recognition server. A screenshot of this testing software is shown in Fig. 8.

First, we tested the speaker recognition module using the Google Speech API. As we explained the difficulty of recognizing 26 machine control commands in Korean in section IV, the recognition results with the Google Speech API only were not excellent. The test accuracies for 26 commands and worker IDs were 89.71% and 78.87% respectively. The Google Speech API often outputs incorrect words when the speech of unusual complex commands is given. Especially, it did not recognize 'gong', another word of zero in Korean, correctly. So, we developed a matching function that finds out the closest command from the recognized output using the Levenshtein distance algorithm [25]. We also made a hard-coded mapping table for frequent wrong outputs. With this post-processing support, the test accuracy has improved to 97.0% overall. While we can get the final output from the Google API in about 0.5 seconds, we had to wait for an additional one second when using the post-processing due to its iterative routines.

Next, the speaker recognition module using our recognition server based on DeepSpeech was tested. Overall, the recognition accuracy was 92.0% for the entire test data set. For 26 control commands, the word error rate in the test data set was 8.7%, but for worker IDs, it dropped to 15.3%. The low recognition accuracy for worker IDs is due to the characteristics of the worker ID data set. A worker ID is composed of 8 digits starting with '2019'. The length of audio data for each digit is very short and there is a lot of imbalance among digits. In order to improve the recognition accuracy, collecting more audio data for training is required. The average recognition time was about 0.23 seconds on the recognition server with a single Intel i7 CPU and a single Nvidia GTX1080ti GPU.

Table 2. The results of recognition

			test accuracy (%)	recognition time (sec)
speech recognition	Google API based	w/o post-processing	84.3	0.50
		w/ post-processing	97.0	1.49
	Deep-Speech baased	commnads (WER)	8.7	0.23
		worker ID (WER)	15.3	
		overall accuracy	92.0	
speaker recognition	SincNet based		99.9	0.06

The speaker recognition result was great. The recognition accuracy was 99.9%, almost perfect, and the recognition time was only 0.06 seconds. These results show that about 100 workers can be identified by voice in a real work environment. We could also guess that the data augmentation technique we used was effective. The results of speech and speaker recognition in the implemented system are summarized in Table. 2.

Ⅵ. Concluding Remark

This paper shows a case study of developing a work management system that uses speech and speaker recognition to improve the working environment. Voice recognition can increase work efficiency by making machine control easier for workers, and speaker recognition improves security by identifying workers with biometric signals. Test results on the implementation of the work management system show that worker recognition by voice is sufficiently applicable in the field. In the case of machine control by speech recognition, there were several points to be supplemented.

If you use the Google Speech API, you can use a reliable speech recognition service for a fee, but post-processing is actually required to match the speech recognition result to the correct control command. Because machine control commands are not terms used in everyday life and there may be high similarities between commands, it is difficult to recognize the command correctly. Such post-processing usually includes iterative routines and hard coding, which can lead to some time delays and difficult automation. When using a speech recognition model such as DeepSpeech through training, post-processing is not required, but it may be expensive because a large amount of training audio data is required to achieve good speech recognition accuracy.

There will be ongoing studies to improve the working environment using voice interfaces. There may also be some other environments where a limited number of spoken sentences must be accurately recognized for specific purposes, such as public safety. We plan to continue research to improve speech recognition accuracy in those environments, using limited amounts of speech data.

References

[1] Google, Cloud Speech-to-Text, see https://cloud.google.com/speech-to-text

[2] Herve Bourlard, Nelson Morgan, Connectionist Speech Recognition: A Hybrid Approach, The Kluwer International Series in Engineering and Computer Science; v. 247, Kluwer Academic Publishers, 1994.

[3] R. Parente, N. Kock, John Sonsini, “An Analysis of the Implementation and Impact of Speech-recognition Technology in the Healthcare Sector.” Perspectives in health information management Vol. 1, 2004.

[4] Kulyukin, V. Human-Robot Interaction Through Gesture-Free Spoken Dialogue. Autonomous Robots 16, pp. 239 - 257 (2004).

[5] Norberto Pires, J. (2005), "Robot by Voice: Experiments on Commanding an Industrial Robot Using the Human Voice", Industrial Robot, Vol. 32 No. 6, pp. 505-511.

[6] Adam Rogowski, Industrially oriented voice control system, Robotics and Computer-Integrated Manufacturing, Elsevier. Vol. 28, Issue 3, June 2012, pp. 303-315.

[7] K. Zinchenko, C. Wu, K. Song, "A Study on Speech Recognition Control for a Surgical Robot," in IEEE Transactions on Industrial Informatics, Vol. 13, No. 2, pp. 607-615, April 2017.

[8] Ismail, Ahmed; Abdlerazek, Samir; El-Henawy, Ibrahim M. 2020. "Development of Smart Healthcare System Based on Speech Recognition Using Support Vector Machine and Dynamic Time Warping" Sustainability 12, No. 6: 2403.

[9] Anwer, Saba; Waris, Asim; Sultan, Hajrah; Butt, Shahid I.; Zafar, Muhammad H.; Sarwar, Moaz; Niazi, Imran K.; Shafique, Muhammad; Pujari, Amit N. 2020. "Eye and Voice-Controlled Human Machine Interface System for Wheelchairs Using Image Gradient Approach" Sensors 20, No. 19: 5510.

[10] Ohneiser, Oliver; Jauer, Malte; Rein, Jonathan R.; Wallace, Matt. 2018. "Faster Command Input Using the Multimodal Controller Working Position “TriControl”" Aerospace 5, No. 2: 54.

[11] Kaczmarek, Wojciech; Panasiuk, Jarosław; Borys, Szymon; Banach, Patryk. 2020. "Industrial Robot Control by Means of Gestures and Voice Commands in Off-Line and On-Line Mode" Sensors 20, No. 21: 6358.

[12] Microsoft, Azure Speech to Text, see https://azure.microsoft.com/en-us/services/cognitive-services/speech-to-text/

[13] Ye-Ji Kim, Yong-Seong Moon, Seong-Hun Jeong, Dae-Han Jeong, Tae-Hyong Kim, “Voice Recognition and Control System Based on Deep Learning for Smart Lighting”, KSC2017, Korea Information Science Society, 2017.12.

[14] Yong-Seong Moon, Ye-Ji Kim, Seong-Hun Jeong, Yu-Hee Kim, Chang-Yeol Lee, Tae-Hyong Kim, “Dialog Management for Voice Recognition based Light Control”, KCC2018, Korea Information Science Society, 2018.06.

[15] C. Shayamunda, T. D. Ramotsoela, G. P. Hancke, "Biometric Authentication System for Industrial Applications using Speaker Recognition," IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, Singapore, 2020, pp. 4459-4464.

[16] E. K. Wang, X. Liu, C. -M. Chen, S. Kumari, M. Shojafar, M. S. Hossain, "Voice-Transfer Attacking on Industrial Voice Control Systems in 5G-Aided IIoT Domain," in IEEE Transactions on Industrial Informatics, 2020. doi: 10.1109/TII. 2020.3023677.

[17] Mozilla, Project DeepSpeech, see https://github.com/mozilla /DeepSpeech, 2016.

[18] Mirco Ravanelli, Yoshua Bengio, “Speaker Recognition from Raw Waveform with SincNet“, arXiv:1808.00158, 2018.

[19] Awni Y. Hannun, Carl Case, J. Casper, Bryan Catanzaro, G. Diamos, Erich Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, A. Ng, "Deep Speech: Scaling up end-to-end speech recognition", arXiv:1412.5567, 2014.

[20] Wit.ai, Inc, wit.ai: Build Natural Language Experiences, see https://wit.ai/

[21] DeepSpeech, “DeepSpeech Model”, https://deepspeech.readthedocs.io/en/v0.9.3/DeepSpeech.html

[22] Mark Heath, NAudio, see https://github.com/naudio /NAudio

[23] librosa, A python package for music and audio analysis, see https://github.com/librosa/librosa

[24] kenlm, KenLM: Faster and Smaller Language Model Queries, see https://github.com/kpu/kenlm.

[25] Wikipedia, Levenshtein distance, see https://en.wikipedia.org /wiki/Levenshtein_distance

Abdulaziz Gaybulayev

2013 Information Science from Tashkent University of Information Technologies (B.S.)
2015 Computer Engineering from Tashkent University of Information Technologies (B.E.)
2018~ Computer Engineering from Kumoh National Institute of Technology. (Ph.D. Candidate)
Field of Interests: Deep Learning, Embedded Systems
Email: g.abdulaziz@kumoh.ac.kr

Jahongir Yunusov

2016 Information Technology from Tashkent University of Information Technologies (B.S.)
2020 Computer Engineering from Kumoh National Institute of Technology. (M.E.)
2020~Divus Corp. (Research Engineer)
Field of Interests: Deep Learning, Computer Vision
Email: jahongir7174@kumoh.ac.kr

Tae-Hyong Kim (김 태 형)

1992 Electronic Engineering from Yonsei University (B.S.)
1995 Electric and Electronic Engineering from Yonsei University (M.S.)
1995 Electric and Electronic Engineering from Yonsei University (Ph.D.)
2002~Computer Engineering from Kumoh National Institute of Technology. (Professor)
Career:
2001 Post-Doc. Researcher, Univ. of Ottawa
2008 Visiting Scholar, Univ. of California, Riverside
Field of Interests: Deep Learning, Big Data Analytics, IoT
Email: g.abdulaziz@kumoh.ac.kr