International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

Enhanced Speech Recognition for Indonesian Geographic Dictionary Using Deep Learning

H. Hugeng, D. Gunawan, A. T. Kusumo

Abstract*: Speech recognition technology has been developing very fast lately. One of its application is to know the meaning of some terms included in a geographic dictionary. When a subject speaks a word to the system, it will output the word and its meaning and explanation. There are many methods that are applied to speech recognition. One of the methods that can be applied and improve the accuracy of speech recognition is the use of a deep learning method, i.e. Convolutional Neural Network (CNN). In this research, CNN's speech recognition accuracy for the Indonesian geographic dictionary is analyzed to show that CNN can improve the accuracy of speech recognition compared to speech recognition with Gaussian mixture model and hidden Markov model (GMM-HMM). CNN is one of deep learning methods that analyzes and finds similarity in Mel-frequency cepstral coefficients (MFCC) from sound waves. This research is performed by making models of the spoken words using CNN under Python and TensorFlow. CNN is trained with these models from speech data collected and prepared from 20 students, consists of 19 men and a woman of different ages from 19 to 23 years. The vocabulary of the database consists of 50 words. The result of this research is a desktop application with the trained models implemented. Our application can recognize well the spoken words from subjects. Testing of the trained models was performed to examine the accuracy of the build speech recognition system. The result of the CNN speech recognition method from the Indonesian geographic dictionary is 80% accuracy for isolated words and 72.67% for continuous words in our research.

Keywords* : Speech Recognition; Convolutional Neural Network; Deep Learning; Indonesian Geographic Dictionary.

I. INTRODUCTION

Automatic Speech Recognition (ASR) has been developed since 1930 to distinguish from simple words to continuous words of people. Interest in ASR has been so great in recent years. This change is due to the sharp increase in demand for ASR in mobile devices and the success of new voice applications in the mobile world, such as Voice Search (VS), Short Message (SMD) and Assistant Virtual Voice (eg Siri from Apple, Google Now and Cortana from Microsoft). Equally important is the development of deep learning (DL) techniques, such as deep neural network (DNN) and convolutional neural network (CNN), which have become more possible and significantly reduce the error rate. This is due to the continuous development of computational power, the availability of a large amount of training data and a better understanding of these two algorithms.

The purpose of ASR is to create transcripts of human speech in spoken words. This is a very difficult task because human speech signals are very different due to speaker attributes, different speech styles, neighborhood noise, and so forth. ASR must assign variable length speech signals in phonetic symbols or variable length rows. As is known, very successful Hidden Markov Models (HMMs) were used with rows of long variation in the handling and temporal behavior of speech signals modeling a number of states, with each state corresponding to a special distribution observation. Gaussian Mixture Models (GMMs) are still the strongest estimate of probability distributions of speech signals considered in each of these states by HMM. Meanwhile, the generative training methods of GMM-HMM for ASR based on the Expectation Maximization (EM) algorithm have been well developed. In addition, numerous training methods used in discrimination [1][2][3] are used together with other HMM ASR systems to refine produce more sophisticated.

In recent years, the HMM models using artificial neural networks (KNN) instead of GMM have generated interest in significant research [4-8], first in the task of telephone recognition TIMIT with monophonic HMM for functions of Keppler coefficient mel frequency (MFCC) [9], and shortly thereafter presented in some ASR assignments with large vocabulary with Triphones HMM models in [10]. In retrospect, the improvements of recent research from the use of "deep" learning, a good indication of the number of hidden layers in the neural network, as well as abstraction and psychological validity of the representations in the most distant layers removed from the post, which was before 30 Drew attention to the attractiveness of KNN for cognitive scientists. No deep learning, the ANN models are very strong discrimination directly the surfaces of each classification can represent feature space without assumptions about the data structure. Instead, the GMM consider each data sample generated from a hidden expert to model a Gaussian and Gaussian-weighted set of components that uses the space's general properties. RNAs have been used for speech recognition for more than two decades. Initial experiments worked on input static and limited speech where fixed size buffers isolate a word in a speech recognition system to handle enough classification information [11][12]. They were used as extractors in continuous speech recognition. Its first success continuous speech recognition application is that it coincides almost exactly with the current use of GMM, i.e. as sources of posterior probabilities of states, given the number of structures fixed features [13].

Manuscript published on 30 September 2019.
*Correspondence Author(s)
Dr. H. Hugeng*, Assoc. Professor, Elec. Eng. Depart., Universitas Tarumanagara, Jakarta, Indonesia.
Dr. D. Gunawan, Professor, Elec. Eng. Depart., Universitas Indonesia, Depok, Indonesia.
T. Kusumo, Comp. Eng., Universitas Multimedia Nusantara, Tangerang, Indonesia.

What is the difference between the hybrid ANN-HMM system and the original approach? This hybrid system is much bigger. The advances in this 20-year-old computer hardware are playing an important role in advancing ANN-based Acoustic Modeling (AM) approaches because ANN training takes place with so many hidden units in today's voice data per hour the day becomes feasible.

The current trend towards the ANN-HMM hybrid system begins with the use of limited Boltzmann machines (RBM), which can take into account the following context. Recent advances in learning by minimizing contrastive divergence [10] allow for an approach to learning. All these factors have a significant effect on performance.

This historical deconstruction is important because the very wide input context and the corresponding representational invariance of domains are crucial to the success of neural network-based acoustic models. An ANN-HMM architecture that embodies these advantages can exceed other ANN architectures with depth unlimited potential for some tasks. CNNs lie between the oldest and most popular deep neural network architecture and have been widely used in handwriting recognition. Different CNN configurations may affect the final performance of ASR.

In the evolution of ASR, CNN has proved to be very good for understanding the speech features. This CNN implementation has been proven by Microsoft Research [14]. In the use of CNN to learn the patterns, data need to be organized before becoming feature maps to be fed as an input to the CNN. Feature maps take a part of an image as a 2-dimension array and move horizontally and vertically based on the image resolution. After feature maps are created, convolution and pooling layers will create a layer that will be called CNN.

Adiyanto has created an Android application of Indonesian geographic dictionary [15]. Unfortunately, this application does not include the speech recognition feature in it. Ferdiansyah and Purwarianti in [16] have made the Indonesian ASR system using acoustic models for Indonesian phonemes derived from English phonemes. There are 29 Indonesian phonemes selected from 39 corresponding English phonemes. They use the Sphinx CMU, the toolkit for ASR. Sound recognition accuracy for 50-words language model (LM) is 70% for isolated words and 73% for continuous words. Our previous research in [17] is an ASR application for Indonesian geographic dictionary. In this research, we use conventional GMM-HMMs to create acoustic models for Indonesian phonemes. The ASR toolkit used is CMU Pocket Sphinx. The ASR accuracy for isolated words is in average of 52.87% in quiet condition and 1 cm microphone distance from the speaker's mouth. This result is worse than [16], so it needs to be improved again.

Similar to the research from Abdel-Hamid et al. in [18], here CNN implementation is made for ASR of Indonesian geographic dictionary. CNN method is state-of-the-art method that is being developed for ASR. The advantages of this method are faster and more accurate compared to GMM-HMMs.

In this research CNN model will be made using Python and Tensorflow[19]. TensorFlow is a framework that accommodates machine learning algorithms. The system in TensorFlow is flexible and used for research in Deep Neural Network models. The dataset that we use limited only to 50 words of Indonesian geographic terms. The input is limited to words and not sentence. We analyze two types of input for this research, isolated words and continuous words. The results of our research show that CNN can increase the accuracy of ASR compared to conventional GMM-HMMs.

II. ALGORITHM OVERVIEW

In this research, we build a program that is able to recognize Indonesian language words using TensorFlow and Python. Due to Rabiner and Juang [20], there are 3 approaches to solve ASR problem, those are acoustic-phonetic approach, pattern recognition approach, and artificial intelligence approach. Our method by using CNN is from artificial intelligence approach. This method is tested using Indonesian geographic terms. CNN uses convolution and pooling layers that differentiate this method against others. CNN model that is used in this research is cnn-trad-fpool3 that is proposed by Sainath and Parada [21]. The input is integer from localized feature as feature maps. The size of a feature maps become small in the upper layer, because of convolution and pooling applied. Usually one or more hidden layers connected is added to the CNN to connecting the feature from all frequency bands before being fed to the output layer.

CNN provides shift invariance in time and space and is critical in achieving up-to-date image recognition performance. It has also been shown to improve the accuracy of speech recognition than some pure DNN tasks. To apply CNN, we must implement several new compute nodes. Also, CNN is proven to improve the accuracy of speech recognition compared to pure deep neural network in some tasks [14][18]. Deep Neural Network – Hidden Markov Model (DNN-HMM) is developing a better performance in recognizing speech compared to GMM-HMM. The reason is because the potency of DNN to model complex correlation in speech feature.

A. Convolutional Layers

According to Yu and Deng [6], convolution is a process that convolve element-wise products of a kernel to an image. An example of a convolution operation is shown in Fig. 1 where the output in the convolution nodes has 2 channels (represented by two 2x2 matrices) and the input has 3 channels (represented by 3 matrices of 3x3 above). A channel is a view of the same image. For example, an RGB image can be displayed with three channels: R, G, and B. Each channel has the same size. In each pair of output and input channel, there is a kernel. The total number of kernels must be the same size with the product of the number of the input channels C_x and the number of output channel C_v.

In Figure 1, $C_x = 3$, $C_v = 2$, and the number of kernels is 6. Every kernel $K_{k\ell}$ from the input channel k and the output channel ℓ is a matrix. The kernel moves over the input with subsampling rate S_r and S_c in row (horizontal) and column (vertical) direction, respectively. For every output channel ℓ and input slice (i,j) (the i^{th} step along the column direction and j^{th} step along the row direction), the evaluation function is as follow:

$$v_{\ell ij}(\mathbf{K}, \mathbf{Y}) = \sum_k \mathrm{vec}(\mathbf{K}_{k\ell}) \circ \mathrm{vec}(\mathbf{Y}_{kij}) \quad (1)$$

where $\mathbf{Y}_{kij}$ has the same size with $\mathbf{K}_{k\ell}$.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

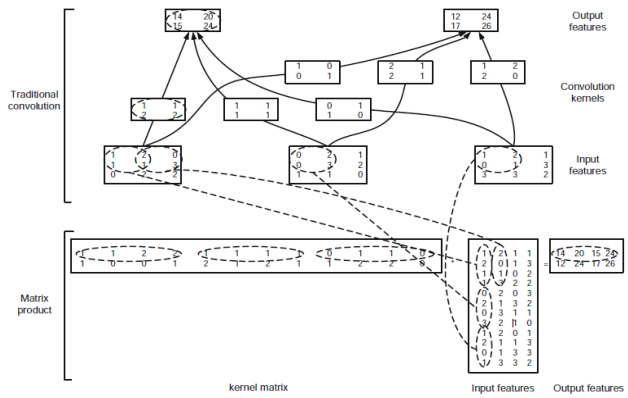

Fig. 1 Example of convolution operations [6]

This evaluation function, $v_{fij}(\mathbf{K}, \mathbf{Y})$, is involving many small matrix operations and can be slow. Chellapilla et al. [22] proposed a technique to convert all these small matrix operations to a large matrix product as shown in Fig.1 bottom. In this way, every kernel parameter can be combined together to a one big matrix **W**. For the convolution node to be used by another convolution node, the conversion needs to be reorganized slightly different from what proposed by [22] by transposing both kernel matrix and the input features matrix as well as the order of the matrices in the product [6]. The result is, each sample in the output can be represented by $O_r \times O_c$ columns of $C_v \times 1$ vectors which can be reshaped to become a single column, where

$$O_r = \begin{cases} \frac{I_r - K_r}{S_r} + 1 & \text{no padding} \\ \frac{(I_r - mod(K_r, 2))}{S_r} + 1 & \text{zero padding} \end{cases} \quad (2)$$

is the number of rows in the output image, and

$$O_r = \begin{cases} \frac{I_c - K_c}{S_c} + 1 & \text{no padding} \\ \frac{(I_c - mod(K_c, 2))}{S_c} + 1 & \text{zero padding} \end{cases} \quad (3)$$

is the number of rows in the output image, where I_r *and* I_c, are number of rows and columns, respectively from the input image. K_r *and* K_c are respectively, number of rows and columns in each kernel. The combined kernel matrix is of size $C_v \times (O_r \times O_c \times C_x)$ and packed feature input matrix is of size $(O_r \times O_c \times C_x) \times (K_r \times K_c)$.

With the conversion, the related computation of the convolution node with operands **W**, **Y** become

$$\mathbf{X}(\mathbf{Y}) \leftarrow Pack(\mathbf{Y}) \quad (4)$$

$$V(\mathbf{W}, \mathbf{Y}) \leftarrow \mathbf{WX} \quad (5)$$

$$\nabla_W^J \leftarrow \nabla_W^J + \nabla_n^J X^T \quad (6)$$

$$\nabla_X^J \leftarrow W^T \nabla_n^J \quad (7)$$

$$\nabla_Y^J \leftarrow \nabla_Y^J + Unpack(\nabla_X^J) \quad (8)$$

This technique enables good parallelization for large matrix operations but applying additional cost to pack and unpack the matrices. In most conditions, the gain is better than the cost. By doing this, we can add bias and nonlinearity to the convolution operations.

B. Pooling Layers

Pooling layers have the purpose of reducing the resolution of feature maps. And there is two common approach of the pooling layers, Max Pooling and Average Pooling.

1) *Max Pooling*: Max Pooling is pooling operation to input **X** inside a window with size K_r x K_c for each channel. The operation window moves along with strides S_r and S_c at the vertical and horizontal direction, respectively. This operation does not change the channel's size, so $C_v = C_x$.

2) *Average Pooling*: Average Pooling is pooling operation that applying average to input **X** inside a window with size K_r x K_c for each channel. Operation window moves along with strides S_r and S_c at the vertical and horizontal direction, respectively.

C. Dataset Collection

As we know that in building a model for machine learning, we need a dataset to train, test, and validate the algorithm used. So at first, we need to collect the speech data of Indonesian geographic terms. This dataset later is fed to the machine to learn using CNN.

Table- I: List of 50 Words from Indonesian Geographic Dictionary

Alloy	Black Hole	Fauna	Humus	Laut
Argon	Bumi	Fosil	Jurang	Magma
Aspal	Dam	Front	Kabut	Magnet
Atlas	Danau	Fungi	Kanal	Menhir
Banjir	Debris	Geyser	Kawah	Orbit
Bank	Debu	Granit	Knot	Palung
Basin	Delta	Gua	Komet	Pulau
Benua	Embun	Gunung	Kompas	Semen
Bigbang	Era	Gurun	Lahar	Tanjung
Bise	Es	Horst	Lalang	Teluk

Dataset collection is needed for making training set that will be used by the application. The words that will be collected are Indonesian geographic terms with a total of 50 words. Dataset is collected from 20 students at our faculty (19 males and 1 female with age varied from 19 – 23 years old) saying those 50 words, as listed in Table I. There are five different samples of each word from each subject. The total amount is 5,000 words of speech data in the dataset. All speech data is recorded by using Audacity application with waveform audio file format (.wav). The format is using 16 bit, sampling rate of 16,000 Hz, and mono audio.

III. RESEARCH METHOD

This research is performed by making the model for the speech recognition using CNN with Python and TensorFlow. CNN is trained using the dataset from the data collection. First, the machine reads all the speech data that has been collected. Then, the machine will give label to each word in the dataset. Every word later will be converted to MFCC that will be fed into the training of CNN.

The result of this research is a desktop application with the trained model implemented. The application can recognize well the words spoken by a subject. After the application is built, we did the testing of the trained model. This research is focused on the accuracy of the application as the parameter of performance. The accuracy test is performed on two conditions, that are isolated words and continuous words.

Most researchers in the world use word error rate (WER) and word accuracy (W. Acc.) to measure the performance of speech recognition application.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

WER is the error recognition from the word that is processed, as given by Eq. 9 below,

$$WER = \frac{S}{H+S} \quad (9)$$

where S is the wrong recognition and H is the right recognition.

In the meanwhile, word accuracy (W. Acc.) is the accuracy rate of the speech recognition. Word accuracy is defined as,

$$W.Acc. = 1 - WER. \quad (10)$$

IV. RESULTS AND DISCUSSION

At first, we need to reorganize the dataset to meet the specification that will be used in training. We need to cut and save each word from the speech using Audacity and save it to WAV format, as described in Fig. 2. We save each spoken word using a file name with the format of "name_nohash_x.wav" to differ one speaker to the others. Nohash is used to recognize that a speech is in the same set with others that has the same name.

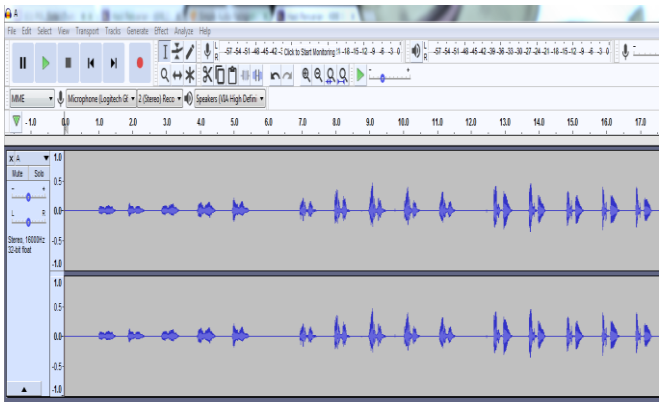

Fig. 1 Example dataset organizing

Later, we organize each word to each folder that has the same name, so that the machine knows and can give label to each data. First, machine will read every single word in each folder and give label. The machine will also know every set inside the folder. After that it will separate training, validation, and test data in the process. The ratio of total data that is used in this research to make the model is 80% training, 10% validation, and 10% testing. Beside these 5,000 spoken words from our data collection, we also add Google Speech Commands dataset to help the machine recognizing unknown words. Only 10% of the data used in the training is unknown.

In the making of this speech recognition, noise factor from background is required to be considered. The goal is to make the application can still recognize the words even if there is a little noise in the speech. We apply some background noise to training dataset. There is also silence data that is used in the training with the amount 10% of the training data. The silence model is made so that the machine can understand silences between adjacent words. After the training is performed, the result of the model is a freezing model. This model is a binary GraphDef file that can be applied to Android, iOS, or Raspberry Pi. Then, the usage of the model can vary, depends on the needs. The total steps of the training are 18,000 steps with 0.001 learning rate.

Isolated words are tested with speech data that are recorded with Audacity. The format is the same with the training data. The speech that has been cut then later be fed to the application. The application then decides which word from the model that is the same with the one being fed to the machine.

Continuous words are tested with PyAudio for real-time recognition of words. This application uses socketio, request, and flask library to accommodate the real-time recognition of words in the input speech. PyAudio will record every input with the same format as the training data. PyAudio will send the speech after a certain threshold is accepted. After there is no sound from the stream, it will save the speech to WAV and fed it to the model. Both isolated words and continuous words are tested six times with different speaker in each test. The speakers are not part of speakers for the training dataset. In each test, the speaker will test 25 random words out of total of 50 words.

Table- II: Accuracy Results of ASR Using CNN

Parameter	Test Type	
	Isolated Words	Continuous Words
Average accuracy	80%	72.67%
Highest accuracy	96%	84%
Lowest accuracy	60%	60%
Average wrong recognitions out of 25 words	5	6.83

As it is shown in Table II, the accuracy result from isolated words is 80% which in average there are 5 wrong recognitions in each test. The highest accuracy in isolated words test is 96% and the lowest is 60%. On the other hand, the accuracy result from continuous words is 72.67% with in average there are 6.83 wrong recognitions in each test. The highest accuracy in continuous words test is 84% and the lowest is 60%.

One of the reasons why the words failed to be recognized may come from the wrong spelling from the speakers. There are some words that have the same sound at the beginning or at the end of the words. One of the words that usually failed to be recognized is "lalang", instead of "lalang" the machine recognized it as "bank" which has the same sound in the end.

As can be seen in Table III, this speech recognition by applying CNN for Indonesia geographic terms outperforms our previous speech recognition by using GMM-HMMs that has only 52.87% accuracy for isolated words in silence and near condition to the speakers [17]. This system is also better than the system of Ferdiansyah and Purwarianti in [16], where they used also GMM-HMMs with average accuracy of 70% for isolated words and 73% for continuous words.

Table- III: Results Comparison Between Methods in Average Accuracy

Methods	Isolated Words	Continuous Words
CNN (this research)	80%	72.67%
GMM-HMMs [16]	70%	73%
GMM-HMMs [17]	52.87%	-

Retrieval Number: K18860981119/19©BEIESP
DOI: 10.35940/ijitee.K1886.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

V. CONCLUSION

From this research, we propose an ASR system of Indonesian geographic dictionary by using CNN. This application of ASR has been made with TensorFlow with vocabulary of 50 terms from Indonesian geographic dictionary. The test of isolated words results in a 7.33% better accuracy than the test of continuous words. There are in average 5 wrong recognitions out of total 25 randomly chosen words in isolated words results, and that is of 6.83 in continuous words results.

Because of the complexity and flexible structure of the CNN or deep learning in general, our experiments explained in Section IV show that CNN outperforms conventional GMM-HMMs method. Our system is better about 10% in average accuracy for isolated words, and about the same performance for the continuous word.

ACKNOWLEDGMENT

The authors gratefully thank to all participating students and colleges in data collection. We also thank to Google, Inc. in providing Google Speech Commands dataset.

REFERENCES

1. H. Jiang, "Discriminative training for automatic speech recognition:A survey," *Comput. Speech, Lang.*, vol. 24, no. 4, pp. 589–608, 2010.
2. X. He, L. Deng, and W. Chou, W., "Discriminative learning in sequential pattern recognition -A unifying review for optimization-oriented speech recognition," *IEEE Signal Process. Mag.*, vol. 25, no. 5, pp. 14–36, 2008.
3. L. Deng and X. Li, "Machine learning paradigms for speech recognition: An overview," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 5, pp. 1060–1089, 2013.
4. G.E. Dahl, M. Ranzato , A. Mohamed, and G.E. Hinton, "Phone recognition with the mean-covariance restricted Boltzmann machine. *Adv. Neural Inf. Process. Syst.*, no. 23, 2010.
5. A. Mohamed, T. Sainath, G. Dahl, B. Ramabhadran, G. Hinton, and M. Picheny, "Deep belief networks using discriminative features for phone recognition," in *Proc. 2011 IEEE ICASSP*, 2011, pp. 5060–5063.
6. D. Yu and L. Deng, *Automatic speech recognition: A deep learning apprpach*, London: Springer-Verlag, 2016.
7. F. Seide, G. Li, X. Chen, and D. Yu, 2011, December. Feature engineering in context-dependent deep neural networks for conversational speech transcription. in *Proc. IEEE Workshop Autom. Speech Recognition Understand. (ASRU)*, 2011, pp. 24–29.
8. N. Morgan, "Deep and wide: Multiple layers in automatic speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 7–13, 2012.
9. A.R. Mohamed, G. Dahl, and G. Hinton, "Deep belief networks for phone recognition," in *Proc. NIPS Workshop Deep Learn. Speech Recognition Related Applications, 2009.*
10. G. Hinton, L. Deng, D. Yu, G.E. Dahl, A.R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
11. Landauer, T.K., Kamm, C.A. and Singhal, S., „Learning a minimally structured back propagation network to recognize speech," in *Proc. 9th Annu. Conf. Cogn. Sci. Soc.*, pp. 531–536, 1987.
12. D. Burr, "A neural network digit recognizer," in *Proc. IEEE Int. Conf. Syst., Man., Cybern., 1986.*
13. H.A. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*, Springer Science & Business Media, 2012, vol. 247.
14. O. Abdel-Hamid, A.R. Mohamed, H. Jiang, and G. Penn, 2012, March. "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *Proc.2012 IEEE ICASSP, March 2012*, pp. 4277-4280.
15. B.R. Adiyanto, "Aplikasi android kamus geografi versi bahasa Indonesia," Presentasi Karya Ilmiah Komputer, Univ. Gunadarma, Depok, Indonesia, 2012.
16. V. Ferdiansyah and A. Purwarianti, "Indonesian automatic speech recognition system using English-based acoustic model," *American Journal of Signal Processing*, vol. 2, no. 4, pp. 60-63, 2012.
17. H. Hugeng and E. Hansel, "Implementation of android based speech recognition for Indonesian geography dictionary," *ULTIMA Computing*, vol. 7, no. 2, pp. 76-82, 2015.
18. O. Abdel-Hamid, A.R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on audio, speech, and language processing*, vol. 22, no. 10, pp.1533-1545, 2014.
19. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, "TensorFlow: A system for large-scale machine learning," in *Proc. the 12th USENIX Symposium on Operating Systems Design and Implementation*, Nov. 2016, vol. 16, pp. 265-283.
20. L.R. Rabiner and B.H. Juang, *Fundamentals of speech recognition*. Englewood Cliffs: PTR Prentice Hall, 1993, vol. 14.
21. T.N. Sainath and C. Parada, "Convolutional neural networks for small-footprint keyword spotting," in *Proc. 16th Annual Conf. of the Int. Speech Communication Association*, 2015, pp. 1478 -1482.
22. K. Chellapilla, S. Puri, and P. Simard, "High performance convolutional neural networks for document processing. in *Proc. 10th Int. Workshop on Frontiers in Handwriting Recognition*, Oct. 2006.

AUTHORS PROFILE

Dr. Hugeng Hugeng is an Associate Professor at Electrical Engineering Department, Universitas Tarumanagara, Jakarta, Indonesia. His current research interest includes digital signal processing, signal processing for communications, image processing, speech recognition, deep learning, and machine learning. He was born in Lubuk Linggau, South Sumatera, Indonesia. He worked as research assistant at RWTH Aachen University, Germany in 2001, and at TU Kaiserslautern, Germany in 2002-2005. He achieved his Ph.D. degree in Electrical Engineering from Universitas Indonesia in 2011. Dr. H. Hugeng has been actively involved in IEEE activities and since September 2016, he is appointed in a key position in IEEE SPS Indonesia Chapter as Technical Activities Coordinator. Dr. H. Hugeng is a senior member of IEEE. He serves as general chair of the 2017 IEEE International Conference on Smart Cities, Automation and Intelligent Computing Systems (ICON-SONICS). Since 2018, he is appointed as chairman of the Tarumanagara International Conference on the Applications of Technology and Engineering (TICATE) where its Proceedings is indexed by Scopus.

Prof. Dr. Dadang Gunawan received the B.Sc. degree in electrical engineering from Universitas Indonesia in 1983, and M.Eng. and Ph.D. degrees from Keio University, Japan, and Tasmania University, Australia in 1989 and 1995, respectively. He is the Head of Telecommunication Laboratory and of the Wireless and Signal Processing Research Group of the Electrical Engineering Department, University of Indonesia. His research interests are wireless communication and signal processing. Prof. Dr. D. Gunawan is a senior member of IEEE and IEEE Signal Processing Society.

Andrew Tirto Kusumo received his B. Eng. degree in computer engineering from Universitas Multimedia Nusantara, Tangerang in 2018. He is an enthusiast in analyzing data and has expertise in software engineering. He worked with PHI-Integration and now he joints JULO, a company for financial technology, as data engineer. His research interest includes deep learning, data analysis, and software engineering.