International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 6, December 2018, pp. 5381~5388
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i6.pp5381-5388

Convolutional Neural Network and Feature Transformation for Distant Speech Recognition

Hilman F. Pardede, Asri R. Yuliani, Rika Sustika
Research Center for Informatics, Indonesian Institute of Sciences, Indonesia

Article Info

Article history:

Received Jan 5, 2018
Revised Jul 27, 2018
Accepted Aug 7, 2018

Keyword:

CNN
Distant Speech Recognition
Feature transformation
LDA
MLLT
Reverberation

ABSTRACT

In many applications, speech recognition must operate in conditions where there are some distances between speakers and the microphones. This is called distant speech recognition (DSR). In this condition, speech recognition must deal with reverberation. Nowadays, deep learning technologies are becoming the the main technologies for speech recognition. Deep Neural Network (DNN) in hybrid with Hidden Markov Model (HMM) is the commonly used architecture. However, this system is still not robust against reverberation. Previous studies use Convolutional Neural Networks (CNN), which is a variation of neural network, to improve the robustness of speech recognition against noise. CNN has the properties of pooling which is used to find local correlation between neighboring dimensions in the features. With this property, CNN could be used as feature learning emphasizing the information on neighboring frames. In this study we use CNN to deal with reverberation. We also propose to use feature transformation techniques: linear discriminat analysis (LDA) and maximum likelihood linear transformation (MLLT), on mel frequency cepstral coefficient (MFCC) before feeding them to CNN. We argue that transforming features could produce more discriminative features for CNN, and hence improve the robustness of speech recognition against reverberation. Our evaluations on Meeting Recorder Digits (MRD) subset of Aurora-5 database confirm that the use of LDA and MLLT transformations improve the robustness of speech recognition. It is better by 20% relative error reduction on compared to a standard DNN based speech recognition using the same number of hidden layers.

Corresponding Author:

Hilman F. Pardede
Research Center for Informatics,
Jl. Cisitu No. 21/154D Bandung, Indonesia.
Email: hilm001@lipi.go.id

1. INTRODUCTION

Deep Learning technologies have recently achieved huge success in acoustic modelling for automatic speech recognition (ASR) tasks [1]-[4]. They replace conventional Hidden Markov Models-Gaussian Mixture Models (HMM-GMM) [5], [6]. Curently, Deep Neural Network (DNN) is the state-of-the-art architecture for speech recognition. DNN is used to provide posterior probability to HMM based on a set of learned features. A hybrid of HMM-DNN has shown to have superior performance compared to HMM-GMM models for ASR.

Currently, more automatic speech recognition (ASR) applications found in our daily activities. They have been implemented as virtual assistant in smart-phones, home automation, meeting diarisation, and so on. For such applications, ASR must operate in conditions where there are some distances between the speakers and the microphones. This is called distant speech recognition (DSR). In such conditions, ASR systems are expected to be robust against noise and reverberation. However, the performance of DNN-HMM

systems are still unsatisfactoryfor these conditions [7]. Noise and reverberation distort the speech signals causing large degradation on the performance of ASR systems. This may hold back the users when using ASR applications.

Many studies have proposed techniques to improve the accuracies of ASR in noisy and reverberant conditions. One approach is to enhance the noisy features by applying noise removal techniques [8] . Others designed a discriminative, handcrafted features that are more robust against noise and reverberation [9]. Many works also propose adapting the acoustic models into noisy condition [10]. In DNN frameworks however, many methods proposed for HMM-GMM systems may not work as well [7]. For deep learning frameworks, various architectures are investigated to find the better systems such as recurrent neural network (RNN) [11] and convolutional neural network (CNN) [12]. In these approaches, the hidden layers of the systems are increased to produce more discriminative features before fine-tuning in the last layers. However, this may significantly increase the computational time for training.

Currently, CNN are gaining interests among researchers. Originally, it is used in computer vision [13]. Some studies [14]-[17] indicate it to be better than DNN for large scale vocabulary tasks. We argues, the properties of CNN such as pooling could benefit in reverberant conditions. In these studies mostly deal with noise only. Their implementations on dealing reverberations have not yet explored.

One advantage of deep learning frameworks is the ability of the network to learn the discriminative features given input data [18]. Studies show that transforming features before feeding them to the networks may benefit the performance of deep learning systems. There are numerous approaches that can be implemented in feature domain to improve the performance of ASR systems in deep architectures for large vocabulary systems. Some of them are linear discriminant analysis (LDA) [19], heteroscedastic linear discriminant analysis (HLDA) [20], Maximum Likelihood Linear Transform (MLLT) [21], feature based-minimum phone error (fMPE) [22], or using the combined transformations.

In this study, we propose CNN with feature transformations for improving the robustness of ASR against reverberation. we apply LDA and MLLT on features before feeding them to CNN. We argue that, applying them may also improve the robustness of speech recognition in reverberant conditions by still using relatively smaller number of hidden layer. We evaluate the use of feature transformations (i.e. LDA and MLLT) on Mel-frequency cepstral coefficient (MFCC) We capture the context information of speech by splicing the features with several preceding and succeeding frames and then applied LDA to reduce the dimensionality. After that, we apply MLLT on the reduced features. In this we feed the transformed features as acoustic input for CNN.

The rest of the paper is organized as follows. Section 2. provide theoretical background for our system. In this section, we briefly describe the features we used, the feature transformations and CNN. In Section 3., we explain our proposed system. In Section 4., we we explain our experimental setup to evaluate our method and discuss the results. We conclude the paper in Section 5.

2. THEORETICAL BACKGROUND

2.1. Speech Features

Many features have been proposed for ASR. MFCC is arguably the most popular one. MFCC is a handcrafted feature that is extracted using two-stages Fourier transform. The aim is applied to decorrelate speech components in time and frequency domains. By doing so, speech units, such as phonemes, could be modeled using mixtures of Gaussians using only their diagonal covariances. In MFCC extraction process, the speech signals are chunked into sequences of frames with fixed duration, usually around 25-50 ms each. Speech is assumed to be stationary for each frames and then the Fourier transform is applied to obtain its spectral components. Usually, the power spectra are used by taking the square of its magnitude. Then, the spectra are mapped into a mel-scaled filter-banks to emphasis the frequency in lower region more. After that, the log operation is applied to the output of mel-filterbank before applying Fourier Transform, in this case only using the real part of Fourier transform to decorrelate each component in frequency domain.

While MFCC shows good results when the conditions between training and testing are the same, it suffers when there is high variability on the data. Speech is highly varied due to intra-speaker variabilities, inter-speaker variabilities, environment variabilities, i.e. when speech is noisy or contaminated, etc. Many studies propose different features to improve the robustness of ASR. PLP is one of the examples. The main difference between PLP and MFCC is that PLP applies cube root function instead of log. The objective is to reduce the sensitivity of the features in low energy region which is most sensitive to noise. Other difference is the use of bark scale instead of mel in MFCC.

MFCC shows pretty good performance in HMM-GMM systems. Since it is quite uncorrelated, it is adequate to model each state of HMM using mixtures of Gaussian only using the diagonal covariances of the GMM. However, two-stages Fourier transform remove the correlation between speech components in time-

frequency domains. These correlations could still be needed in recognition process. This may be one of the reason that ASR are not robust. In DNN-HMM systems, some studies show it is more benefit when more "raw" features are used. One of them is FBANK [23]. FBANK has the same extraction process with MFCC except it is without the second stage Fourier transform. So, some correlations in frequency are still exist.

2.2. Feature Transformation

Transforming features to other domain spaces often found effective in many classification tasks in machine learning. The use of high dimension features are often ineffective because it may lead to overfitting. Reducing the dimensions of the features is often applied. LDA and PCA are examples of feature reduction techniques. LDA [24] is applied in supervised manner while PCA is an unsupervised technique.

LDA is usually applied in preprocessing stage to reduce the dimensions of feature from the n-dimensional features are reduced into m-dimensional space (m<n).The objective is to project the features space into lower dimensional space and making the features more discriminative. The lower dimensional feature space is chosen such that it emphasizes the distances between classes more than within the class. Mathematically it could be written:

$$J(\theta) = \frac{\det\left(\theta \Sigma_i \theta^T\right)}{\det\left(\theta \Sigma_e \theta^T\right)} \tag{1}$$

where Σ_i are the covariance between class, Σ_e is the covariance within class, θ is the feature, and $J(\theta)$ is the cost function that to be maximized. The solution for $J(\theta)$ is by taking the first m eigenvectors of matrix Σ_e^{-1} Σ_i after sorting the eigenvalues from the largest ones. For more information on applying LDA on speech features could refer to [19]

Meanwhile, MLLT [25] is applied in HMM-GMM systems to loosen the assumption in HMM-GMM systems. In HMM-GMM systems, it is assumed that the features are independent with each other. Therefore, Gaussian assumptions are with only diagonal co-variances are used. While this could fasten the training time, the assumption may not necessarily holds. This is because speech component may still be related to each other in feature space. MLLT [25], which is also known as semi-tied co-variance (STC) [26], linearly transforms the sample data to a new transformed space that are Gaussian distributed to loosen this assumption. MLLT is applied to implicitly capture the correlation between the feature elements by using constrained co-variance model.

MLLT works as follow. MLLT uses eigen decomposition to decompose a full co-variance matrix over the set of Gaussian components, and each components maintain its "diagonal" characteristics. A full covariance matrix could be decomposed using the following formula:

$$\hat{\boldsymbol{\Sigma}}^{(m)} = \mathbf{H}^{(r)} \hat{\boldsymbol{\Sigma}}_{diag}^{(m)} \mathbf{H}^{(r)T} \tag{2}$$

where m is the index of Gaussian component and r is the index of class. Each component m has three parameters: weight, mean, and diagonal element of semi-tied co-variance matrix. So, each co-variance matrix $\boldsymbol{\Sigma}^{(m)}$ could be decompose into two: diagonal element of co-variance matrix component m, $\boldsymbol{\Sigma}_{\boldsymbol{diag}}^{(\boldsymbol{m})}$, and a shared full co-variance matrix of Gaussian components in class r, $\mathbf{H}^{(r)}$ (named as semi-tied transform). We denote $\mathbf{A}^{(r)}$ be the inverse of $\mathbf{H}^{(r)}$. In ASR, the covariance matrices are trained under Maximum Likelihood sense on the training data and will be optimized with respect to $\widehat{\mathbf{A}}^{(r)}$ the mean of the Gaussians $\mu^{(m)}$ and diagonal covariance matrices $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{diag}}^{(\boldsymbol{m})}$. So, the cost function J could be written:

$$J\left(\aleph, \hat{\aleph}\right) = \sum_{m \in M^{(r)}, \tau} \gamma_m(\tau) \log\left(\frac{\left|\hat{\mathbf{A}}^{(r)}\right|^2}{\left|\operatorname{diag}\left(\hat{\mathbf{A}}^{(r)} \mathbf{W}^{(m)} \hat{\mathbf{A}}^{(r)T}\right)\right|} \right) \tag{3}$$

where:

$$\mathbf{W}^{(m)} = \frac{\Sigma_\tau \gamma_m(\tau) \left(\mathbf{o}(\tau) - \hat{\mu}^{(m)}\right)\left(\mathbf{o}(\tau) - \hat{\mu}^{(m)}\right)^T}{\Sigma_\tau \gamma_m(\tau)} \tag{4}$$

$$\beta = \sum_{m \in M^{(r)}, \tau} \gamma_m(\tau) \tag{5}$$

and

$$\gamma_m(\tau) = p\left(q_m(\tau)|\aleph, \mathbf{O}_T\right) \tag{6}$$

The notation $q_m(\tau)$ is the gaussian component m at time τ, $\mathbf{O}_T$ is the training data and $\mathbf{o}(\tau)$ is the observed feature. Then the maximum likelihood estimation of the mean is:

$$\hat{\mu}^{(m)} = \frac{\Sigma_\tau \gamma_m(\tau)\mathbf{o}(\tau)}{\Sigma_\tau \gamma_m(\tau)} \tag{7}$$

and the covariance matrix estimate is:

$$\hat{\mathbf{\Sigma}}_{diag}^{(m)} = \text{diag}\left(\mathbf{A}^{(r)}\mathbf{W}^{(n)}\mathbf{A}^{(r)T}\right) \tag{8}$$

Calculating $\mathbf{A}^{(r)}$ is nontrivial. To estimate it, it is initialized using an identity matrix and then estimate $\widehat{\mathbf{\Sigma}}_{\boldsymbol{diag}}^{(\boldsymbol{m})}$ using Equation (8) and then $\mathbf{A}^{(r)}$ is updated using Eg. (2)

2.3. Convolutional Neural Network

A typical convolutional network structure is illustrated in Figure 1. This is different from DNN, where all neurons in the previous layers are connected to all the neurons of the successive layers, which may not be effective when the features have large dimensions. Convolutional Neural Network (CNN) is a special kind of deep neural network. CNN introduces two types of special network layers, called convolutional layer and pooling layer. Each neuron of the convolutional layer receives inputs from a set of filters of the lower layer. The filters are obtained by multiplying a small local part of the input with the weight matrix, where these filters are then replicated throughout the whole input space. Localized filters that share the same weight appear as feature maps. After completing convolution process, a pooling layer takes inputs from a local part of the convolutional layer and generates a lower resolution version of filter activation.

In the implementations for speech recognition, after few layers of CNN structured, a fully connected layer of generative deep neural network model (DBN-DNN) is performed to combine extracted local patterns from all positions in the lower layer for final recognition [27]. In this paper, we use two layers of CNN structure and then apply 4 layers of DBN to produce a total of 6 layers of hidden layers for pretraining. Then DNN is applied on the top of then for supervised learning.

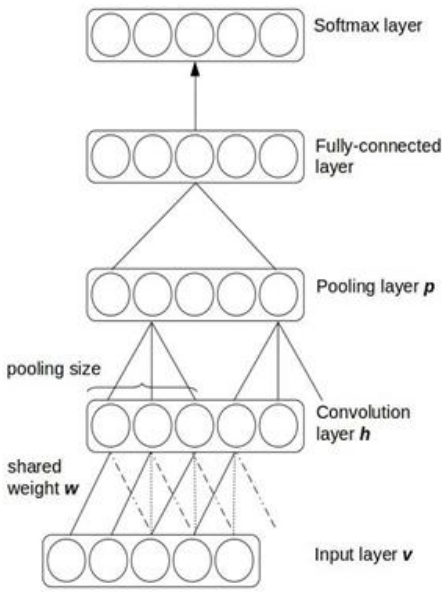

Figure 1. A typical CNN architecture

3. THE PROPOSED SYSTEM

Figure 2 is the currently the most commonly used ASR systems. DBN-DNN, which is based on Karel's implementation of DNN on KALDI [28] is used as baseline. DBN configuration in this experiment uses 6-depth hidden layers with dimension of 2048 hidden neurons, i.e. Gaussian-Bernoulli RBM as first layer connecting to the Gaussian acoustic inputs and Bernoulli-Bernoulli RBM layers afterward. The stack of pre-training layers is followed by DNN layers with 1 hidden layers (1024 neurons) and softmax output layer. We denote this as BASELINE2 in this paper.

We also train a conventional HMM-GMM. We denote this as BASELINE1. For this, we model each digit with 16 states HMM, left-to-right where each state was modelled using Mixtures of Gaussian with the number of Gaussian is three. For pause models: sil, we use HMM with 3 states with 6 Gaussian components.

Figure 3 is the proposed system in this study. We use MFCC as the basic for static features. We use 13 dimensions of static features and then the features are spliced by using 4 preceding and succeeding frames to capture the context of the speech producing 117 dimensions. Then, we apply LDA to reduce the dimensions into 40 dimensions for all features. Then we apply MLLT on the output of LDA before feeding them into CNN. For systems using only LDA, we denote as PROPOSED1, and for system with both LDA and MLLT is denoted as PROPOSED2.

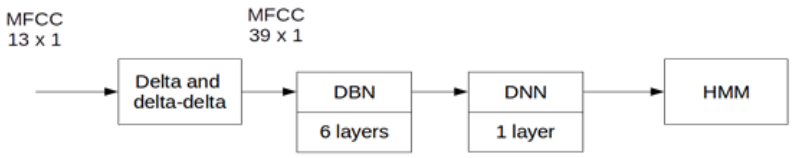

Figure 2. The Baseline System: MFCC with delta transformation before feeding to DBN-DNN

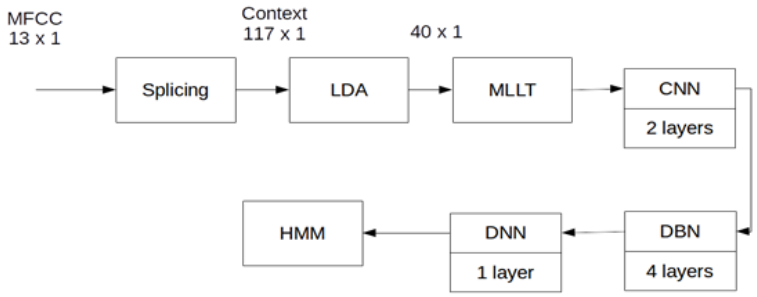

Figure 3. The Proposed System: The Feature Transformation (LDA and MLLT) is applied on MFCC before feeding it into 2-layers CNN. The output of CNN are feed into 4 layers DBN and 1 layer DNN

For CNN pre-training, we use two layers of CNN and then 4 layers of DBN. For CNN layers, we use 128 neurons for first hidden layer and 256 neurons for second hidden. Pool size of three and max pooling are used in this study. For DBN, we use standard 1024 neurons for hidden layers. This settings are the same as in [15] as it is found a good setting for speech recognition. The output of DNN is used to estimate the posterior probability of HMM states in hybrid of deep learning and HMM systems.

4. EXPERIMENTS

4.1. The Setup

The experiments are evaluated on speech corpus of isolated digit recognition task. We use TIDigits corpus to train acoustic models on clean conditions, which consists of 8623 utterances pronounced by 111 male and 114 female adult speakers. For test data, we use the reverberant version of TIDigits, that is the Meeting Recorder Digits (MRD) subset of Aurora-5 corpus [29]. The corpus comprises of real recording in hands-free mode in the meeting room. The speech data is collected from 24 speakers at the International Computer Science Institute in Barkeley, resulting of 2400 utterances for each microphone. The recording is performed using four microphones (labeled as 6, 7, E, and F) which are placed at the middle of the table. The

recording thus contain some reverberant acoustic conditions from the effect of hands-free recording in a meeting room. The performance is measured using word error rate (WER).

4.2. Results and Discussion

Table 1 shows the evaluation of the proposed methods (PROPOSE1 and PROPOSE2). As comparison, the performance of BASELINE1 and BASELINE2 are shown as well. The table clearly indicates a consistent reduction of word error rates in reverberant conditions. PROPOSED1 achieves 37.67 % and 27.78 % relative improvements over BASELINE1 and BASELINE2 respectively while PROPOSED2 achieves 38.69% and 28,94 % relative improvements over BASELINE1 and BASELINE2 respectively. Applying MLLT after LDA (PROPOSED2) slightly better than applying LDA alone.

This might be analysed as the followings. When reverberation exists, the resulting reverberant speech are a sum of the signal with the delayed version of the same signals. Therefore reverberant speech contain the information from previous frames. This will increase correlations between neighboring frames, hence may increase the local correlations nearby frames. In CNN architecture, the properties of locality, convolution and pooling may be benefit in such conditions. Since the emphasis is on local neuron first, it can learn on the local information and produce good features based on clean part of speech from early frames (since they are relatively clean compared to late part of speech). When speech is corrupted by reveberation, It may cause some frequency shift (delay in time-frequency domains).

These delays are difficult to handle within other models such as GMMs and DNNs, where many Gaussians and hidden units are needed to be optimized for all possible pattern shifts [27]. With pooling properties in CNN, the same feature value that calculated from different location is collected together and indicated by a single value, which may be from the cleaner part of speech. Therefore, the differences in features extracted by poling ply may minimized the effect of delay which are caused by reverberation. LDA finds the features with the large variances and most separated means within the class. So, when it is used for features, it is very likely to choose most distinguish spectra (the dominant spectra) which may contain the phonemes information. When CNN is applied, due to the max-pooling, the information is maintained up to the top layers, producing a more discriminative features and hence improving the performance.

Table 1. WER (%) of the Proposed Method in Comparison with the Baselines

Model	Conditions					Average
	Clean	MRD 6	MRD 7	MRD E	MRD F	
BASELINE1	0.64	46.66	54.56	50.20	44.59	49.00
BASELIINE2	0.80	39.93	47.99	42.69	38.51	42.28
PROPOSE1	0.90	28.10	33.90	33.03	27.14	30.54
PROPOSE2	0.82	27.73	33.20	32.62	26.61	30.04

5. CONCLUSION

In this study, we evaluate the use of LDA and MLLT on CNN-based speech recognition to improve the robustness of speech recognition against reverberation. Our experiments confirm that our proposed method is more robust than standard DNN-HMM and HMM-GMM systems. The properties of weight sharing, pooling, and locality of CNN, could improve the recognition accuracy on all transformed features compared to the standard fully-connected DNN.

We need to state that the evaluated tasks are digit recognition tasks. Therefore, the long-term dependency that exists in speech may not as significant as in continuous speech. Therefore it is interesting to see how each architecture fare for continuous tasks. Since reverberation time is also heavily influenced by the size of the room, it is also interesting to see how deep architectures perform in different settings of rooms. This is our future plan.

REFERENCES

[1] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* 2013, pp. 7398–7402.

[2] T. Yoshioka and M. J. Gales, "Environmentally robust asr front-end for deep neural network acoustic models," *Computer Speech & Language*, vol. 31, no. 1, pp. 65–86, 2015.

[3] R. Errattahi and A. El Hannani, "Recent advances in lvcsr: A benchmark comparison of performances," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 6, pp. 3358–3368, 2017.

[4] M. F. Alghifari, T. S. Gunawan, and M. Kartiwi, "Speech emotion recognition using deep feedforward neural network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 10, no. 2, pp. 554–561, 2018.

[5] K. F. Akingbade, O. M. Umanna, and I. A. Alimi, "Voice-based door access control system using the mel frequency cepstrum coefficients and gaussian mixture model," *International Journal of Electrical and Computer Engineering*, vol. 4, no. 5, p. 643, 2014.
[6] S. N. Endah, S. Adhy, and S. Sutikno, "Comparison of feature extraction mel frequency cepstral coefficients and linear predictive coding in automatic speech recognition for indonesian," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 15, no. 1, pp. 292–298, 2017.
[7] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 7398–7402.
[8] P. Lockwood and J. Boudy, "Experiments with a nonlinear spectral subtractor (nss), hidden markov models and the projection, for robust speech recognition in cars," *Speech Communication*, vol. 11, no. 2, pp. 215 – 228, 1992.
[9] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (pncc) for robust speech recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 24, no. 7, pp. 1315–1329, Jul. 2016.
[10] M. J. F. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination," *IEEETransactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 352–359, Sep 1996.
[11] C. Weng, D. Yu, S. Watanabe, and B. H. F. Juang, "Recurrent deep neural networks for robust speech recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2014, pp. 5532–5536.
[12] Y. Zhang, W. Chan, and N. Jaitly, "Very deep convolutional networks for end-to-end speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2017, pp. 4845–4849.
[13] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE transactions on neural networks*, vol. 8, no. 1, pp. 98–113, 1997.
[14] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional neural networks for distant speech recognition," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1120–1124, 2014.
[15] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8614–8618.
[16] T. N. Sainath, B. Kingsbury, A.-r. Mohamed, G. E. Dahl, G. Saon, H. Soltau, T. Beran, A. Y. Aravkin, and B. Ramabhadran, "Improvements to deep convolutional neural networks for lvcsr," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2013, pp. 315–320.
[17] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition," in 2012 *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 4277–4280.
[18] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
[19] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *1992 IEEE International Conference on Acoustics, Speech and Signal Processing*, 1992, pp. 13–16.
[20] L. Burget, "Combination of speech features using smoothed heteroscedastic linear discriminant analysis." in *Interspeech*, 2004.
[21] R. A. Gopinath, "Maximum likelihood modeling with gaussian distributions for classification," in *1998 IEEE International Conference on Acoustics, Speech and Signal Processing*, 1998, pp. 661–664.
[22] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fmpe: Discriminatively trained features for speech recognition," in *2005 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2005, pp. I–961.
[23] T. Yoshioka, A. Ragni, and M. J. Gales, "Investigation of unsupervised adaptation of dnn acoustic models with filter bank input," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 6344–6348.
[24] S. Geirhofer, "Feature reduction with linear discriminant analysis and its performance on phoneme recognition," *Department of Electrical and Computer Engineering: University of Illinois at Urbana-Champaign*, 2004.
[25] M. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," C*omputer Speech & Language*, vol. 12, no. 2, pp. 75 – 98, 1998.
[26] M. J. Gales, "Semi-tied covariance matrices for hidden markov models," I*EEE transactions on speech and audio processing*, vol. 7, no. 3, pp. 272–281, 1999.
[27] A.-H. Ossama, M. Abdel-rahman, J. Hui, D. Li, P. Gerald, and Y. Dong, "Convolutional neural networks for speech recognition," *IEEE Signal Processing Magazine*, vol. 22, no. 10, 2014.
[28] K. Vesel`y, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks." in *INTERSPEECH*, 2013, pp. 2345–2349.
[29] H. Hirsch, "Aurora-5 experimental framework for the performance evaluation of speech recognition in case of a hands-free speech input in noisy environments," *Niederrhein Univ. of Applied Sciences*, 2007.

BIOGRAPHIES OF AUTHORS

Hilman Pardede is a researcher at Research Center for Informatics, Indonesian Institute of Sciences. He obtained his Bachelor Degree in Electrical Engineering from University of Indonesia in 2004 and Master of Engineering from the University of Western Australia in 2009. He received his Doctor of Engineering from Tokyo Institute of Technology in 2013. He did a postdoctoral at Fondazione Bruno Kessler in Trento Italy from 2013 to 2015. His research interests include are speech recognition, pattern recognition, signal processing, machine learning and artificial intelligence. He is an IEEE member and reviewer for Speech Communications (Elsevier) and International Journal of Machine Learning and Cybernetics (Springer). He also has served as reviewers in several international conferences.

Asri Rizki Yuliani is a researcher at Research Center for Informatics, Indonesian Institute of Sciences. She earned bachelor degree in Computer Science from the University of Teknologi Malaysia in 2009 and master degree in Information Management from Yuan Ze University in 2013. Her research interests include speech recognition, pattern recognition, and machine learning.

Rika Sustika is a researcher at Research Center for Informatics, Indonesian Institute of Sciences (LIPI). She earned bachelor and master degree in Electrical Engineering from Bandung Institute of Technology (ITB). Her research interests are in the area of signal processing. Since January 2017 joined with machine learning research group. Interested for using deep learning on many application such as on speech recognition, image recognition, and natural language processing.