

I-vectors and Deep Convolutional Neural Networks for Language Identification in Clean and Reverberant Environments

Panikos Heracleous¹, Yasser Mohammad^{1,2}, Kohichi Takai¹, Keiji Yasuda¹,
Akio Yoneyama¹

¹KDDI Research, Inc., Japan

2-1-15 Ohara, Fujimino-shi, Saitama, 356-8502, Japan

{pa-heracleous,ko-takai,ke-yasuda,yoneyama}@kddi-research.jp

²Artificial Intelligence Research Center, AIST, Japan

yasserm@aun.edu.eg

Abstract. In the current study, a method for automatic language identification based on deep convolutional neural networks (DCNN) and the i-vector paradigm is proposed. Convolutional neural networks (CNN) have been successfully applied to image classification, speech emotion recognition, and facial expression recognition. In the current study, a variant of typical CNN is being applied and experimentally investigated in spoken language identification. When the proposed method was evaluated on the NIST 2015 i-vector Machine Learning Challenge task for the recognition of 50 in-set languages, a 3.9% equal error rate (EER) was achieved. The proposed method was compared to two baseline methods showing superior performance. The results obtained are very promising and show the effectiveness of using DCNN in spoken language identification. Furthermore, in the current study, a front-end feature enhancement and dereverberation approach based on a deep convolutional autoencoder is also reported.

Keywords: Spoken language identification, deep convolutional neural networks, dereverberation, denoising autoencoder

1 Introduction

Automatic spoken language identification is the process of automatically recognizing language in a spoken utterance. Language identification is an important part of speech-to-speech translation systems and has a significant role in the diarization of meetings. Moreover, it can be utilized in call centers to automatically route incoming calls to appropriate native speaker operators.

Several studies have investigated spoken language identification. The approaches presented here are categorized based on the features they employ. Language identification systems are categorized into the following approaches: acoustic-phonetic, phonotactic, prosodic, and lexical [1]. In phonotactic systems [1, 2], sequences of recognized phonemes obtained from phone recognizers

are modeled. In [3], a typical phonotactic language identification system is used, where a language-dependent phone recognizer is followed by parallel language models (PRLM). In [4] a universal acoustic characterization approach to spoken language recognition is proposed. The main idea is to describe any spoken language with a common set of fundamental units, such as manner and articulation, which are used to build a set of language-universal attribute models. The vector space modeling-based phonotactic language recognition approach is demonstrated in [1, 5] and presented in [6]. The key idea is to vectorize a spoken utterance into a high-dimensional vector, thus leading to a vector-based classification problem.

In acoustic modeling-based systems, however, each recognized language is modeled by using different features. Although significant improvements in LID have been achieved from phonotactic approaches, most state-of-the-art systems still rely on acoustic modeling.

In [7], an early attempt at language identification based on a neural network is presented. Similarly, neural network-based language identification is addressed in [8]. In [9] the first attempt at language identification using deep learning is presented. In [10] automatic language identification based on deep neural networks (DNN) is also presented. This method demonstrates performance superior to i-vector-based [11] classification schemes when a large amount of data is used. The method is compared to linear logistic regression, linear discriminant analysis- (LDA) based, and Gaussian modeling-based classifiers. When limited training data are used, the i-vector yields the best identification rate. Another method based on DNN and using deep bottleneck features is presented in [12]. A method for identification from short utterances based on long short-term memory (LSTM) recurrent neural networks (RNN) is presented in [13]. In [14] the problem of language identification is addressed by using i-vectors with support vector machines (SVMs) [15] and LDA. SVM with local Fisher discriminant analysis is also used in [16]. Similarly to the current study, the method is evaluated on the NIST 2015 i-vector Machine Learning Challenge task. The results obtained closely resemble the results obtained in the current study when using SVM. In [17] deep neural networks-based language identification is also presented. The method is also evaluated on the NIST 2015 i-vector Machine Learning Challenge task.

In the current study, a method for automatic language identification based on the i-vector paradigm and deep convolutional neural networks (DCNN) is proposed. Convolutional neural networks [18, 19] have been successfully applied to sentence classification [20], image classification [21], facial expression recognition [22], and in speech emotion recognition [23]. Furthermore, in [24] bottleneck features extracted from CNN are used for robust language identification.

The motivation of using CNN, instead of the conventional fully connected feed-forward neural network, lies in the fact that CNN require less parameters. As a result is cheaper in terms of memory and compute power compared to DNN. Furthermore, previous studies reported robustness against noise of CNN. Also, since the inputs to the network are i-vectors and not sequences, CNN offer

a reliable solution to the classification problem. In the current study, DCNN, a variant of typical CNN, is used and experimentally investigated in spoken language identification.

In addition to language identification, the effectiveness of convolutional neural networks in dereverberation is also addressed. Specifically, a convolutional denoising autoencoder (DAE) [25] is used to map the reverberant speech into clean speech. DAEs are used for feature enhancement using a class of deep neural networks (DNN) and they are trained to map a corrupted speech observation to a clean one. DAEs have been successfully used in automatic speech recognition in noisy and reverberant environments, and they have been shown to improve recognition rates significantly.

2 Methods

2.1 Data

In the NIST 2015 LRE i-Vector Machine Learning Challenge task, i-vectors, constructed from conversational and narrow-band broadcast speech, are given as training, testing, and development data. The task covers 50 languages, and contains 15000 training i-vectors, 6500 test i-vectors, and 6431 development i-vectors. The training i-vectors are extracted from speech utterances with a mean duration of 35.15s. The training data and the test data are labeled, but the development i-vectors are unlabeled. The set also includes i-vectors corresponding to out-of-set languages. In the current study, only the in-set languages are considered. In particular, 300 training i-vectors and 100 test i-vectors are used for each of the 50 in-set languages.

In the current study, *NTT-AT multilingual speech database for telephony 1994* was also used to investigate the effectiveness of deep convolutional denoising autoencoders in de-reverberation. The data cover 21 languages and four male and four female speakers are assigned to each language. Twenty-four short utterances (i.e., approximately 4 sec) are spoken by each native speaker. Speech data are sampled at 16-bit and 16kHz rates.

The reverberant emotional speech data are simulated based on the convolution method. Specifically, impulse responses are recorded in different environments and then convoluted with the clean data in order to create the reverberant emotional speech data. For recording, a linear microphone array with 14 transducers located at 2.83cm intervals is used [26]. The impulse response is measured using the TSP method [27]. TSP length is 65536 points and the number of synchronous additions is 16. Impulse responses in five different rooms are recorded. The $T_{[60]}$ reverberation times are 0.30, 0.47, 0.60, 0.78, and 1.3 seconds, respectively.

2.2 The i-vector Paradigm

Gaussian mixture models (GMM) with universal background models (UBM) are widely used for speaker recognition. In such a case, each speaker model is

created by adapting the UBM using maximum a posteriori (MAP) adaptation. A GMM supervector is constructed by concatenating the means of the adapted model. Similar to speaker recognition, GMM supervectors can also be utilized for language identification.

The main disadvantage of GMM supervectors is the high dimensionality, which requires high computation and memory costs. In the i-vector paradigm, the limitations of high dimensional supervectors (i.e., concatenation of the means of GMMs) are overcome by modeling the variability contained in the supervectors with a small set of factors. Considering automatic language identification, an input utterance can be modeled as:

$$\mathbf{M} = \mathbf{m} + \mathbf{T}\mathbf{w} \quad (1)$$

where \mathbf{M} is the language-dependent supervector, \mathbf{m} is the language-independent supervector, \mathbf{T} is the total variability matrix, and \mathbf{w} is the i-vector. Both the total variability matrix and language-independent supervector are estimated from the complete set of the training data.

2.3 Classification Approaches

Support Vector Machines (SVM) A support vector machine is a discriminative classifier, which is widely used in regression and classification. Given a set of labeled training samples, the algorithm finds the optimal hyperplane, which categorizes new samples. SVM is among the most popular machine learning methods. The advantages of SVM include the support of high-dimensionality, memory efficiency, and versatility. However, when the number of features exceeds the number of samples the SVM performs poorly. Another disadvantage is that SVM is not probabilistic because it works by categorizing objects based on the optimal hyperplane.

Probabilistic Linear Discriminant Analysis (PLDA) PLDA is a popular technique for dimension reduction using the Fisher criterion. Using PLDA, new axes are found, which maximize the discrimination between the different classes. PLDA was originally applied to face recognition [28], and is applied successfully to specify a generative model of the i-vector representation. PLDA was also used in speaker recognition. Adapting to emotion recognition, for the i -th emotion, the i-vector $\mathbf{w}_{i,j}$ representing the j -th recording can be formulated as:

$$\mathbf{w}_{i,j} = \beta + \mathbf{S}\mathbf{x}_i + \mathbf{e}_{i,j} \quad (2)$$

where β is a global offset (i.e., mean of training vectors), \mathbf{S} represents the between-emotion variability, and the latent variable \mathbf{x} is assumed to have a standard normal distribution, and to represent a particular emotion and channel. The residual term $\mathbf{e}_{i,j}$ represents the within-emotion variability, and it is assumed to have a normal distribution with zero mean and covariance $\mathbf{\Sigma}$.

After the training and test i-vectors are computed, PLDA is used to decide whether two i-vectors belong to the same class. For this task, a test i-vector

and an emotion i-vector are required. The emotion i-vectors are computed as the average of the training i-vectors, which belong to a specific emotion. A classification trial requires the emotion i-vectors, the test i-vector, and the PLDA model $\{\beta, \mathbf{S}, \Sigma\}$ parameters.

Convolutional Neural Networks (CNN) A deep neural network is a feedforward neural network with more than one hidden layer. The units (i.e., neurons) of each hidden layer take all outputs of the lower layer and pass them through an activation function. A convolutional neural network is a special variant of the conventional network, which introduces a special network structure. This network structure consists of alternating convolution and pooling layers.

2.4 Evaluation Measures

In the current study, the EER (i.e., equal false alarms and false rejections), the identification rates, and the cost functions are used as evaluation measures. Considering that in the current study only the in-set languages are being recognized, the cost function defined by NIST is modified as follows:

$$C_{avg} = \frac{1}{n} \sum_{k=1}^n P_{error}(k) \cdot 100 \quad (3)$$

where

$$P_{error}(k) = \frac{\text{No. of errors for class } k}{\text{No. of trials for class } k} \quad (4)$$

The identification rate is defined as:

$$acc = \frac{1}{n} \sum_{k=1}^n \frac{\text{No. of corrects for class } k}{\text{No. of trials for class } k} \cdot 100 \quad (5)$$

where n is the number of the target languages. In addition, the detection error tradeoff (DET) curves, which show the function of miss probability and false alarms, are also given.

The effectiveness of the convolutional denoising autoencoder in feature enhancement is evaluated using the mean squared error (MSE) and cosine similarity values. The MSE is defined as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2 \quad (6)$$

The cosine similarity is given by the following formula:

$$\cos(\theta) = \frac{\sum_{i=1}^n Y_i \hat{Y}_i}{\sqrt{\sum_{i=1}^n Y_i^2} \sqrt{\sum_{i=1}^n \hat{Y}_i^2}} \quad (7)$$

Table 1: C_{avg} cost for different language sub-set of NIST LRE 2015.

No of Languages	Classification method		
	DCNN	SVM	PLDA
10	4.2	5.4	8.2
20	9.6	10.7	16.5
30	11.7	13.9	23.6
40	13.8	15.7	27.9
50	16.0	18.6	32.9

Table 2: EER for different language sub-set of NIST LRE 2015.

No of Languages	Classification method		
	DCNN	SVM	PLDA
10	1.8	2.4	3.2
20	3.7	4.1	5.5
30	3.6	4.5	6.0
40	3.4	4.7	6.3
50	3.9	5.2	6.7

where $\hat{\mathbf{Y}}$ is the vector of n predictions, and \mathbf{Y} is the input vector, which generated the predictions.

3 Results

3.1 Language Identification Experiments

These sections present the experimental results for automatic language identification. The experimental results show the performance of the proposed method compared to SVM, and PLDA for the identification of 10, 20, 30, 40, and 50 in-set target languages using the NIST 2015 LRE i-Vector Machine Learning Challenge task. For language identification, a DCNN with four convolutional layers, one *Maxpooling* layer, and one fully connected output layer was used. The number of filters in the convolutional layers was set to 32, 128, 128, and 128, respectively, and the epochs number was set to 150. In the convolutional layers, the *ReLU* activation function was used, and in the fully connected output layer, the *Softmax* activation function was chosen. Finally, the dropout probability was set to 0.15.

Table 1 shows the costs for 10, 20, 30, 40, and 50 target languages, respectively. The results show that the lowest average costs are obtained when using DCNN. It is also shown, the DCNN is followed by SVM. The results show the effectiveness of the proposed method in spoken language identification. In the case of the 50 in-set target languages, the average cost function is only 16.0%, which is a very promising result and superior to the results obtained from other

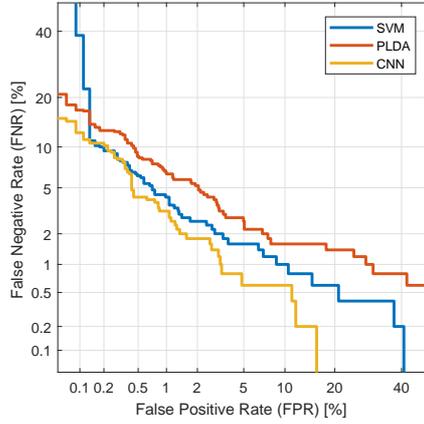


Fig. 1: DET curves for ten target languages.

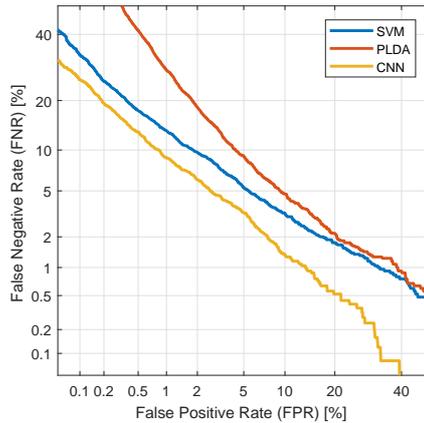


Fig. 2: DET curves for the fifty target languages.

similar studies. The results also show that the cost when using PLDA rapidly increases as the number of target languages increases, and that PLDA is less effective for the current task.

Table 2 shows the EER when using the five sub-set target languages. As shown, when using the DCNN approach, the lowest EER is obtained, followed by SVM. The EER when using PLDA is the highest among the three classifiers. As the table shows, when DCNN is used, the EERs in the case of 20, 30, 40, and 50 languages are very similar. This result indicates the robustness of DCNN in different sub-set target languages.

Table 3: Equal error rates using training data of different sizes.

No. of training i-vectors	Classification method		
	DCNN	SVM	PLDA
2500	6.2	8.3	7.2
5000	4.8	6.7	6.9
7500	4.4	6.4	7.2
10000	4.3	6.1	7.6
12500	4.0	5.8	7.3
15000	3.9	5.2	6.7

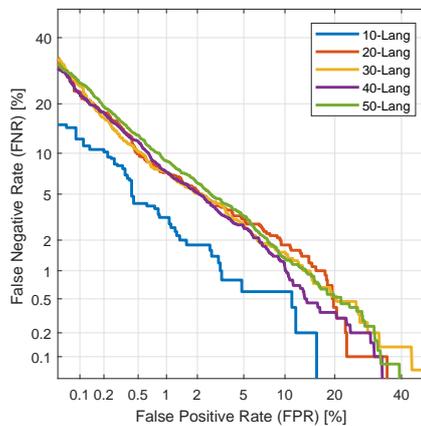


Fig. 3: DET curves for different sub-set target languages using DCNN.

Figure 1 and Fig. 2 show the DET curves in the case of 10 and 50 target languages, respectively. The figures make it clear that superior performance is obtained when the proposed deep CNN-based approach is used.

To investigate the effect of the training data size when using the three classifiers, an experiment is conducted using reduced training data. Table 3 shows the EER obtained in this case. The results show that using only 50 training i-vectors for each target language, the EER is still as low as 6.2%. As shown, the DCNN classifier shows the lowest EER, and it is followed by SVM. Figure 3 shows the DET curves for different sub-set target languages when using DCNN. As shown, differences can be obtained in the case where 10 target languages were used. In all other cases, the DET curves are very similar.

3.2 Front-end Feature Enhancement Experiments

This section presents the results when convolutional denoising autoencoder was used for feature enhancement. Twelve mel-frequency cepstral coefficients (MFCC)

Table 4: Mean squared errors (MSE) for speech dereverberation.

$T_{[60]}$ [sec]	Denoising method		
	Reverberant	CNNDAE	DAE
0.30	1.64252	0.569070	0.570243
0.47	2.101042	0.569152	0.568758
0.60	2.346825	0.568656	0.569388
0.78	1.874870	0.568518	0.568780
1.30	1.701246	0.568803	0.572073

Table 5: Cosine similarities for speech dereverberation.

$T_{[60]}$ [sec]	Denoising method		
	Reverberant	CNNDAE	DAE
0.30	0.858569	0.960958	0.960942
0.47	0.789376	0.960941	0.960928
0.60	0.763129	0.960859	0.960889
0.78	0.824312	0.960880	0.960807
1.30	0.842408	0.961047	0.960879

were extracted every 10 ms using a window of 25 ms. The MFCCs were then concatenated with shifted delta cepstra (SDC) coefficients to form feature vectors of size 112. The inputs to the convolutional denoising autoencoder (CNNDAE) are the feature vectors extracted from clean and reverberant speech, respectively. The encoder part of the CNNDAE consists of two convolutional layers and two *MaxPooling* layers. Each of the *MaxPooling* layers performs compression in the half-dimension. The decoder part consists of three convolutional layers and two *UpSampling* layers. The proposed method was also compared to a method based on a typical, fully connected denoising autoencoder (DAE) with one hidden layer and 32 units. Table 4 shows the MSEs for reverberant speech and dereverberant speech. As shown, in most of cases CNNDAE shows slightly lower MSEs compared to DAE. The MSE values show the effectiveness of using a deep convolutional denoising autoencoder in speech dereverberation. On the other hand, the results show that the performance when using CNN and DAE denoising autoencoders is closely comparable. Table 5 shows the cosine similarities. Similar to MSE values, when using CNNDAE the highest similarities are being obtained.

4 Conclusion

In this study, we proposed a method for language identification based on i-vectors and deep convolutional neural networks. The method was evaluated on the NIST 2015 LRE i-Vector Machine Learning Challenge task and demonstrated

performance that is superior to that obtained using SVM and PLDA classifiers. For the identification of the 50 in-set languages, a 3.9% EER and a 16.0% cost were obtained. Using SVM, a 5.2% EER and 18.6% cost were achieved. Furthermore, a method for dereverberation in language identification was proposed. The proposed method is based on convolutional denoising autoencoder, and its effectiveness in speech dereverberation was demonstrated. As future work, spoken language identification experiments in reverberant environments based on a convolution denoising autoencoder will be conducted.

References

1. Haizhou Li, Bin Ma, and Kong Aik Lee, "Spoken Language Recognition: From fundamentals to Practice," in *Proc. of the IEEE*, vol. 101, no. 5, pp. 1136–1159, 2013.
2. M. A. Zissman, "Comparison of Four Approaches to Automatic Language Identification of Telephone Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4(1), pp. 31–44, 1996.
3. D. Caseiro and I. Trancoso, "Spoken Language Identification Using the Speechdat Corpus," in *Proc. of ICSLP'98*, 1998.
4. S. M. Siniscalchi, J. Reed, T. Svendsen, and C.-H. Lee, "Universal Attribute Characterization of Spoken Languages for Automatic Spoken Language Recognition," *Computer speech and language*, vol. 27, pp. 209–227, 2013.
5. C.-H. Lee, "Principles of Spoken Language Recognition," in *Springer Handbook on Speech Processing and Speech Communication*, J. Benesty, Y. Hunag. M. M. Sondhi, Editors, SpringerVerlag, 2008.
6. Douglas A. Reynolds, William M. Campbell, Wade Shen, and Elliot Singer, "Automatic Language Recognition Via Spectral and Token Based Approaches," in *Springer Handbook on Speech Processing and Speech Communication*, J. Benesty, Y. Hunag. M. M. Sondhi, Editors, SpringerVerlag, 2008.
7. R. Cole, J. Inouye, Y. Muthusamy, and M. Gopalakrishnan, "Language Identification With Neural Networks: a Feasibility Study," in *Proc. of IEEE Pacific Rim Conference*, pp. 525–529, 1989.
8. M. Leena, K. Srinivasa Rao, and B. Yegnanarayana, "Neural Network Classifiers for Language Identification Using Phonotactic and Prosodic Features," in *Proc. of Intelligent Sensing and Information Processing*, pp. 404–408, 2005.
9. G. Montavon, "Deep Learning for Spoken Language Identification," in *NIPS workshop on Deep Learning for Speech Recognition and Related Applications*, 2009.
10. I. L.-Moreno, J. G.-Dominguez, O. Plchot, D. Martinez, J. G.-Rodriguez, and P. Moreno, "Automatic Language Identification Using Deep Neural Networks," in *Proc. of ICASSP*, pp. 5337–5341, 2014.
11. N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19(4), pp. 788–798, 2011.
12. B. Jiang, Y. Song, S. Wei, J.-H. Liu, I. V. McLoughlin, and L.-R. Dai, "Deep Bottleneck Features for Spoken Language Identification," *PLoS ONE*, vol. 9(7), pp. 1–11, 2010.
13. R. Zazo, A. L.-Diez, J. G.-Dominguez, D. T. Toledano, and J. G.-Rodriguez, "Language Identification in Short Utterances Using Long Short-Term Memory (LSTM) Recurrent Neural Networks," *PLoS ONE*, vol. 11(1): e0146917., 2016.

14. N. Dehak, P. A.T.-Carrasquillo, D. Reynolds, and R. Dehak, "Language Recognition via Ivectors and Dimensionality Reduction," in *Proc. of Interspeech*, pp. 857–860, 2011.
15. N. Cristianini and J. S.-Taylor, "Support Vector Machines," *Cambridge University Press, Cambridge*, 2000.
16. P. Shen, X. Lu, L. Liu, and H. Kawai, "Local Fisher Discriminant Analysis for Spoken Language Identification," in *Proc. of ICASSP*, pp. 5825–5829, 2016.
17. S. Ranjan, C. Yu, C. Zhang, F. Kelly, and J. H. L. Hansen, "Language Recognition Using Deep Neural Networks With Very Limited Training Data," in *Proc. of ICASSP*, pp. 5830–5834, 2016.
18. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. 2012, pp. 1097–1105, Curran Associates, Inc.
19. O. Abdel-Hamid, A.-R. Mohamed, H. D. 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, pp. 1533–1545, 2014.
20. Y. Kim, "Convolutional Neural Networks for Sentence Classification," in *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, 2014.
21. W. Rawat and Z. Wang, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review," *Neural Communication*, vol. 29, pp. 23522449, 2017.
22. X.-P. Huynh, T.-D. Tran, and Y.-G. Kim, "Convolutional Neural Network Models for Facial Expression Recognition Using BU-3DFE Database," in *Information Science and Applications (ICISA) 2016. Lecture Notes in Electrical Engineering*, K. Kim and N. Joukov, Eds., vol. 376, pp. 441–450. Springer, 2013.
23. W. Lim, D. Jang, and T. Lee, "Speech Emotion Recognition Using Convolutional and Recurrent Neural Networks," in *Proc. of Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016.
24. S. Ganapathy, K. Han, S. Thomas, M. Omar, M. V. Segbroeck, and S. S. Narayanan, "Robust Language Identification Using Convolutional Neural Network Features," in *Proc. of Interspeech*, 2014.
25. T. Ishii, H. Komiyama, T. Shinozaki, Y. Horiuchi, and S. Kuroiwa, "Reverberant Speech Recognition Based on Denoising Autoencoder," in *Proc. of Interspeech*, pp. 3512–3516, 2013.
26. S. Nakamura, K. Hiyane, F. Asano, and T. Endo, "Sound Scene Data Collection in Real Acoustical Environments," *J. Acoust. Soc. Japan (E)*, vol. 20, No.3, 1999.5.
27. Y. Suzuki, F. Asano, H. Y. Kim, and T. Sone, "An Optimum Computer-Generated Pulse Signal Suitable for the Measurement of Very Long Impulse Responses," *J. Acoust. Soc. Am.*, vol. Vol. 97, No. 2, pp. 1119–1123, 1995.
28. S.J.D. Prince and J.H Elder, "Probabilistic Linear Discriminant Analysis for Inferences About Identity," In *Proc. of International Conference on Computer Vision*, pp. 1–8, 2007.