HARMONIC FEATURE FUSION FOR ROBUST NEURAL NETWORK-BASED ACOUSTIC MODELING

Osamu Ichikawa[1], Takashi Fukuda[1], Masayuki Suzuki[1], Gakuto Kurata[1], Bhuvana Ramabhadran[2]

[1]Watson Multimodal, IBM, Tokyo 103-8510, Japan
[2]Watson Multimodal, IBM, Yorktown Heights, NY 10598, USA
[1]{ichikaw, fukuda1, szuk, gakuto}@jp.ibm.com, [2]bhuvana@us.ibm.com

ABSTRACT

Acoustic modeling with deep learning has drastically improved the performance of automatic speech recognition (ASR) where the main stream of the acoustic feature is still log-Mel filtered one. While the log-Mel filtered features lose harmonic-structure information, they still include useful information for ASR. Several attempts have been made to integrate higher-resolution information into the network. In order to improve the ASR accuracy in noisy conditions, we propose new features integrated into acoustic modeling to represent which parts in the time-frequency domain have a distinct harmonic structure, since it is partially observed in noisy environments. The new features are combined with the standard acoustic features, and the network is trained with them using various noisy data. Through these operations, it learns the acoustic features with a kind of quality tag describing which parts are clean or degraded. Our model reduced the word error rate in an Aurora-4 task by 10.3% in DNN compared with the strong baseline while retaining the high accuracy in clean test cases.

Index Terms— *harmonic structure*, data augmentation, feature fusion, acoustic model, neural networks

1. INTRODUCTION

To avoid data sparseness, machine learning typically used lower dimensional features. In acoustic modeling, 13 dimensional Mel-frequency cepstral coefficients (MFCCs) (with including their delta and delta-delta) have been prevalent for a long time along with Gaussian mixture models (GMMs), and 24 to 40 dimensional log-Mel filtered features have become default in acoustic models with deep neural networks (DNNs) and convolutional neural networks (CNNs). In general, almost all commercial-level automatic speech recognition (ASR) decoders accept low-resolution log-Mel filtered features or MFCC features for inputting where the Mel-filter-bank removes higher-resolution information in the speech signal. Typically, the harmonic structure in human speech is lost.

Several attempts have been made to integrate higher-resolution information into acoustic modeling. The simplest form is to input high-resolution spectrum into the network without Mel-filter bank [1][2]. Larger dimension of Mel-filter-bank was also investigated [3][4]. Multiple-resolution approaches have also been researched [5][6]. Deep scattering spectrum (DSS) designed structural wavelet filters to capture information embedded in a high-resolution spectrum and the modulation [7]. A data-driven approach has also been explored. A filter-bank layer was placed in the network and trained with higher-resolution input [8]. One step further, the input can be raw speech data (wave form) by integrating front-end capability to be trained in the network [1][9][10][11][12]. These approaches focus on integrating the fine structure observed in high resolution speech data. Therefore, the fine structure can be anything local. It does not have to be a frequency-wise periodic structure, but they fully rely on the training in the design of the filters capturing local structures.

In contrast, the technique proposed in this paper captures periodic structures explicitly; through high-resolution cepstrum operations. That information constructs separate features accounting only for a harmonic structure. More specifically, it is designed to indicate the density of the observed harmonic structure in each Mel-frequency band. As shown in Fig. 1, in very noisy situations, harmonic structures are often lost and partially kept, possibly around the formant frequencies of vowels. These bands with the partial harmonic structure should have more speech power and be regarded as more reliable than others.

To extract such information, we introduced local-peak-weight (LPW) coefficients generated by the cut-off operation of a high-resolution cepstrum in our previous paper [13]. For our first step, the coefficients were applied for speech enhancement in combination with the re-trained acoustic model. Next, they were Mel-filtered as the Mel-LPW features for voice activity detection (VAD) modeling [14]. Then, the Mel-LPW coefficients were processed with a sigmoid function to produce a frequency-wise confidence metric for the model-based noise reduction [15].

However, the LPW-based features for acoustic model were not successfully applied. Its simple integration into the GMM-based acoustic model just resulted in poor performance. A part of the reason is that the LPW itself does not include sufficient information to discriminate phones. However, the focus on acoustic modeling has shifted to deep learning, which is more flexible in the mathematical assumptions than GMM. With deep learning, we have a higher chance for successfully integrating such

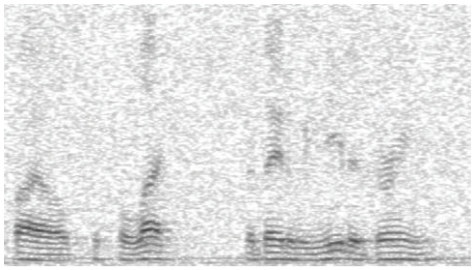

Fig. 1. An example spectrogram in a noisy environment.

metric-type features.

This paper describes harmonic features (HFs) integrated in CNN and DNN in combination with standard acoustic features. HFs are obtained from Mel-LPW coefficients after special normalization. In the training process, we also introduce data augmentation [16] so as to increase noise variation in the training data, because the network need to learn how the standard acoustic features are altered in the noisy conditions in association with the HF information. In that sense, our technique shares a concept similar to noise embedding [17][18] in the context of noise awareness. However, HFs focus on frequency-wise confidence rather than on describing noise characteristics. Also, they have the advantage that they can be seamlessly integrated in CNN, because they share the same spectro-temporal space with the standard features.

2. HARMONIC FEATURE

First, we briefly review how the LPW coefficient is calculated. In [13], LPW was fully described as an LPE-filter. It extracts the harmonic structure partially observed in the noisy spectrum in each frame. The unique aspect is that it does not use F0 (pitch frequency) explicitly, which is often difficult to detect accurately in noisy environment. Instead, it extracts periodic fluctuations in the assumed range of F0, such as 80 to 300Hz.

The observed log power spectrum $Y_t(j)$ is converted to a cepstrum $C_t(i)$ by using $D(i,j)$, a Discrete Cosine Transformation (DCT) matrix. The index t is a frame number and j is the bin number of the DFT. Note this is not a Mel-filtered cepstrum but a high resolution cepstrum having 256 or 128 dimensions.

$$C_t(i) = \sum_j D(i,j) \cdot Y_t(j) \cdot \quad (1)$$

Then, the lower and upper cepstra should be filtered out.

$$\hat{C}_t(i) = \varepsilon \cdot C_t(i) \quad \text{if } i < c_{min} \text{ or } i > c_{max} \quad (2)$$

$$\hat{C}_t(i) = C_t(i) \quad \text{otherwise,} \quad (3)$$

where ε is very small constant. The range of the cepstra (between c_{min} and c_{max}) is chosen to cover the standard F0 range in the human voice.

The filtered cepstrum $\hat{C}_t(i)$ is converted back to a log power spectrum by using an inverse DCT.

$$W_t(j) = \sum_i D^{-1}(j,i) \cdot \hat{C}_t(i) \cdot \quad (4)$$

It is further converted back to a linear power spectrum domain to obtain the LPW coefficient as w.

$$w_t(j) = \exp(W_t(j)) \cdot \quad (5)$$

We obtain Mel-LPW as $\hat{w}$ by processing with the Mel-filter bank as

$$\hat{w}_t(d) = \sum_j w_t(j) \cdot B(d,j) \Big/ \sum_{j'} B(d,j'), \quad (6)$$

where $B(d,j)$ is the d-th triangle filter for the j-th bin.

HFs are obtained after the normalization of the Mel-LPW coefficients. For standard acoustic features, global and local (per utterance or per speaker) normalization of the mean and variance are typically used in deep learning. However, we found they did not work well for HFs. Because HF has small variances in higher frequency bands where ordinary variance normalization unnecessarily amplifies the component. Therefore, selecting the appropriate normalization for HFs is critical to show their advantage. We recommend two kinds of normalization described in the following sub-sections.

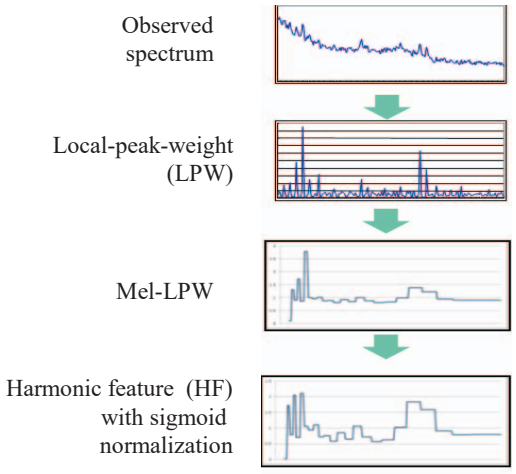

Fig. 2. Process to generate harmonic feature.

2.1. Sigmoid normalization

This normalization compresses the dynamic range from zero to one with the sigmoid function as

$$v_t(d) = 1.0/(1.0 + \exp(-a \cdot (\hat{w}_t(d) - 1.0 - b))) \cdot \quad (7)$$

v is the generated HF. It is the same as the frequency-wise confidence metric introduced in [15]. Unlike ordinary normalization, it does not refer to any global or local statistics. a and b are constant values. Fig. 2 shows the process-flow with an example of the data.

2.2. Global mean and max-variance normalization

In this option, the Mel-LPW coefficient is converted to logarithmic variable $\tilde{w}$. Then, the global mean and variance are calculated for each band with the whole training data.

$$\tilde{w}_t(d) = \log(\hat{w}_t(d) + \varepsilon), \quad (8)$$

$$m(\mathrm{d}) = E[\tilde{w}_t(\mathrm{d})], \quad (9)$$

$$\sigma^2(d) = E\left[(\tilde{w}_t(d) - m(d))^2\right], \quad (10)$$

where $E[\ ]$ takes the expectation. Unlike ordinary variance normalization, we pick the largest variance of all bands and normalize with the values.

$$\sigma_{\max} = \max_d(\sigma(d)) \cdot \quad (11)$$

$$v_t(d) = (\tilde{w}_t(d) - m(d))/\sigma_{\max} \cdot \quad (12)$$

v is the generated HF. By sharing the same variance across the bands, this operation can be regarded as scaling rather than equalizing variations. Similar to sigmoid normalization, it does not refer to local statistics.

3. INTEGRATION OF HARMONIC FEATURE

3.1. DNN

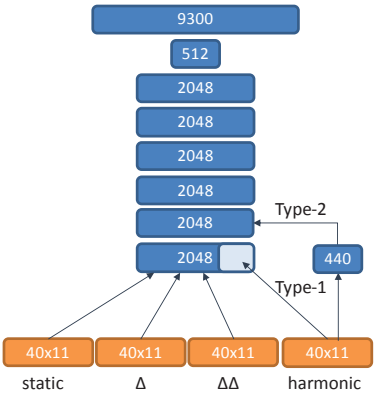

Fig. 3. Harmonic feature with DNN.

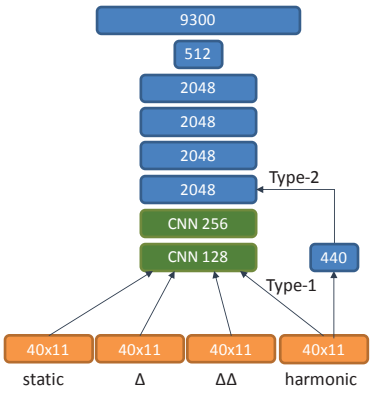

Fig. 4. Harmonic feature with CNN.

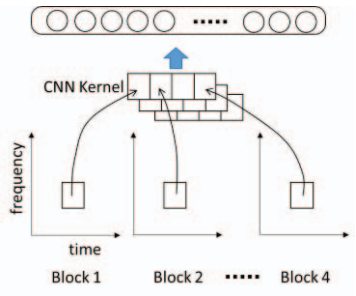

Fig. 5. Integration of input-blocks.

As shown in Fig. 3, the proposed HFs are combined with standard acoustic features such as log-Mel filtered features with their delta and delta-delta. We used a 40-dimensional Mel-filter bank, and the features were spliced with five frames before and after in the input. The output layer corresponds to context-dependent phones.

In general, the first layer of the DNN is considered to be signal-processing-type operation. However, the characteristics of HF are much different from the standard acoustic features. Having an interaction between them in an upper layer would be better. However, in this paper, we still focus on the benefit in our existing framework using standard DNN and CNN. That is Type-1 in Fig. 3.

In Type-1, we optionally introduced a special initialization technique that splits the connections into two blocks. A part of the first layer is reserved only for HFs. The connections across the blocks are set to zero. This is done during the initialization time only, expecting an interaction between HFs and the standard acoustic features that may occur more in the upper layers through training. In this paper, we refer to it as "block initialization".

Type-2 connecting HFs to a middle layer was also explored.

3.2. CNN

Because HFs share the same spectro-temporal space with the standard acoustic features, they can be integrated into CNN without changing the network topology. As shown as Type-1 in Fig. 4, HFs are input to CNN as an additional block.

In CNN, the localized convolutions in each block are combined together, as shown in Fig. 5. Therefore, HFs interact with the standard acoustic features locally in the spectro-temporal space. Our expectation in Type-1 is that HFs may help to specialize CNN kernels for various noise conditions.

Additionally, Type-2 connecting HFs to a full connection layer of CNN model was investigated.

4. AUGMENTATION

Our proposed technique shows advantage only when the acoustic model is trained with noisy data, because the network should learn how the features are transformed in noisy environments with HF as a kind of environmental descriptor. Therefore, if the training data is clean, data-augmentation is recommended.

Augmentation includes convolution of room impulse responses (RIRs) followed by noise addition, so as to increase acoustic variations in the training data. As we used switchboard data for the training, there were very little variation in noise and channel characteristics without it. We applied three kinds of augmentation randomly selected per utterance. They are 1) clean (no-augmentation), 2) in-car data, and 3) REVERB challenge data. Data 2) is our collected data including noise recorded in various driving situations and RIR measured in a luxury sedan. Data 3) includes various RIRs and noise in the multi-condition training set [19]. Therefore, 2) and 3) involved another random selection of RIRs and noise. It was also performed per utterance.

5. EXPERIMENTS

We evaluated the benefits of HFs with the acoustic models trained with and without them.

5.1. Training setup

The network topology is shown in Fig. 3 for DNN and Fig. 4 for CNN. Both networks had 7 hidden layers including one bottle-neck layer. They had sigmoid activation functions. The output layer had softmax units that corresponded to the context-dependent HMM states of penta-phone. The drop-out was not introduced. The CNN model used two convolutional layers [20] in addition to five fully connected layers. All of the 128 nodes in the first feature-extracting layer were attached with 9x9 filters that were two dimensionally convolved with the input log-Mel representations. The second feature extracting layer with 256 nodes had a similar set of 3x4 filters that processes the non-linear activations after max-pooling from the preceding layer. The nonlinear outputs from the second feature-extracting layer were then passed onto the subsequent fully connected layers.

The training data consisted of 300 hours of the Switchboard English conversational telephone speech task recorded at 8 kHz sampling rate. The speech data were pre-processed with the augmentation described in Section 4. Then, the data were coded into a spectrum of 20-ms frame size with a 10-ms frame shift. They were further processed into log-Mel filtered feature with a 40-dimensional Mel-filter bank. They were normalized with the global mean and variance, followed by utterance-based mean normalization. Before inputting to the networks, their delta and delta-delta features were generated and they were expanded in the temporal context of 11 frames. This procedure was for the standard features. For HFs, please refer to Section 2. In the training, they

were fully randomized, and the networks were trained with stochastic gradient descent on mini-batches of 250 frames and a cross-entropy criterion.

5.2. Decoding setup

The trained networks were used in a hybrid decoding scenario [21]. Softmax in the output layer was removed and the logarithmic output score was combined with priors. The decoding language model was a general purpose 4-gram LM with 250k vocabulary commonly used for the test sets.

The objective of this evaluation is to show the noise robustness of our technique without any side-effects in clean conditions. We used Aurora-4 for evaluating in noisy conditions. NIST Hub5 '00 and ASpIRE (single microphone) [22] were used to test the performance in clean and reverberant conditions. Please note 16 kHz audio data were down-sampled to 8 kHz in the evaluation of Aurora-4 and ASpIRE.

5.3. Experimental results and discussion

Our baseline models in DNN and CNN were trained only with the standard features, while our proposed models were trained with HFs as well. The augmentation was applied in all the models, except the reference model (CNN 0).

Table 1 shows the detailed Aurora-4 results with DNN models. WV2 data were recorded with a different microphone from the one used for WV1 and noise data. Our models of DNN 2, 3 and 5 outperformed the baseline model of DNN 1 overall. They showed significant gains in all of the noisy cases in Aurora-4. We believe such a unanimous win endorses the advantages of HFs are general, not specialized for a certain environment. In sigmoid normalization, constants *a* and *b* were 5.0 and 0.3 respectively. They were not tuned, but just imported from our previous research [15]. The DFT size for HFs was set to 512, because it performed better than in 256 for sigmoid normalization. DNN 5 reduced the number of errors by 10.3% overall. Table 2 shows the summary of all the three test-sets. DNN2, 3 and 5 models did not make any drawbacks in the reverberant condition (ASpIRE) and the clean condition (Hub5 '00), and showed even visible improvement. DNN 4 is a model for reference that was built without block initialization. It underperformed in all the test sets comparing with DNN 3. These results support the importance of block initialization for Type-1. In Type-1, sigmoid normalization worked better than max-variance. The best performance was gained by Type-2 (DNN 5).

Table 3 shows a summary of all the three test sets for the CNN model performance. CNN 1 is our baseline model and CNN 0 is for the reference without augmentation. The augmentation significantly improved noisy and reverberant conditions with small draw-backs in the clean condition. Because we focused on noisy conditions, CNN 1 was selected for our baseline model. Our models of CNN 2, 3 and 4 significantly improved the performance in the noisy conditions (Aurora-4) compared with that of the baseline model of CNN 1. The error reduction rate by CNN 4 was 7.7%. CNN 2, 3 and 4 models did not have any drawbacks in the reverberant condition (ASpIRE) and the clean condition (Hub5 '00). The relative improvement by our technique was less in CNN than DNN. However, because CNN is more advantageous than DNN for acoustic modeling in general, the best results were made by CNN. Type-2 model (CNN 4) showed the best performance in all the test-sets. Unlike DNN models, max-variance normalization worked better than sigmoid normalization in clean and reverberant conditions with Type-1 CNN models.

6. CONCLUSION

We presented the integration of harmonic features into DNN and CNN in combination with the standard acoustic features. Unlike other high resolution approaches, our harmonic features represent where the harmonic structure is distinct. Because they can be interpreted as a frequency-wise confidence metric, the network can learn how the acoustic features degrade in noisy conditions in association with them. Our best model showed a constant improvement for all noise types in Aurora-4, with a small improvement in clean and reverberant conditions of Hub5 '00 and ASpIRE as well.

Table 1. Detailed results of Aurora-4 test-set, in word error rate (%). Bold font indicates it is better than the baseline of DNN 1

	model	DNN 1	DNN 2	DNN 3	DNN 5
	augmentation	yes	yes	yes	yes
	HF (norm.)	no	sigmoid	max-var	sigmoid
	Block Init.	-	yes	yes	-
	Topology	-	Type-1	Type-1	Type-2
WV1	clean	7.0	7.0	7.2	7.1
	airport	17.2	**16.3**	**16.8**	**15.5**
	babble	18.9	**18.3**	**18.3**	**17.6**
	car	9.7	**8.8**	**9.0**	**8.9**
	restaurant	22.6	**19.5**	**21.0**	**18.7**
	street	21.0	**19.2**	**20.6**	**19.7**
	train	19.8	**17.9**	**18.4**	**18.4**
WV2	clean	9.0	**8.7**	**8.9**	**8.6**
	airport	24.5	**22.0**	**23.6**	**22.0**
	babble	27.0	**24.7**	**25.0**	**24.5**
	car	12.7	**11.1**	**12.1**	**11.5**
	restaurant	29.1	**26.1**	**28.6**	**25.5**
	street	27.9	**25.4**	**26.3**	**24.6**
	train	26.0	**22.3**	**24.0**	**22.2**
Average		19.5	**17.7**	**18.5**	**17.5**

Table 2. Summary of the three tasks' results with DNN models. Bold font indicates it is better than the baseline of DNN 1

model	DNN 1	DNN 2	DNN 3	DNN 4	DNN 5
augmentation	yes	yes	yes	yes	yes
HF (norm.)	no	sigmoid	max-var	max-var	sigmoid
Block Init.	-	yes	yes	no	-
Topology	-	Type-1	Type-1	Type-1	Type-2
Aurora-4	19.5	**17.7**	**18.5**	**18.6**	**17.5**
ASpIRE	48.5	**47.9**	**48.3**	48.9	**47.7**
Hub5 '00	17.0	**16.5**	**16.8**	**17.0**	**16.2**
Average	28.3	**27.4**	**27.9**	**28.2**	**27.1**

Table 3. Summary of the three tasks' results with CNN models. Bold font indicates it is better than the baseline of CNN 1.

model	*CNN 0*	CNN 1	CNN 2	CNN 3	CNN 4
augmentation	no	yes	yes	yes	yes
HF (norm.)	no	no	sigmoid	max-var	sigmoid
Topology	-	-	Type-1	Type-1	Type-2
Aurora-4	*24.7*	18.3	**17.4**	**17.7**	**16.9**
ASpIRE	*59.2*	47.7	**47.4**	**46.7**	**46.4**
Hub5 '00	*15.4*	16.7	**16.6**	**16.3**	**15.9**
Average	*33.1*	27.6	**27.1**	**26.9**	**26.4**

12. REFERENCES

[1] Z Tüske, P Golik, R Schlüter and H Ney, "Acoustic modeling with deep neural networks using raw time signal for LVCSR," *in Proc. Interspeech*, 2014.

[2] T. N. Sainath and B. Li, "Modeling time-frequency patterns with LSTM vs. convolutional architectures for LVCSR tasks," *in Proc. Interspeech*, 2016.

[3] A. Mohamed, G. Hinton and G. Penn, "Understanding how deep belief networks perform acoustic modelling," *in Proc. ICASSP*, 2012.

[4] H. Sak, A. Senior, K. Rao and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *Computing Research Repository*, *arXiv,* 1507.06947, 2015.

[5] J. Chen, Y. Wang and D. Wang, "A feature study for classification-based speech separation at low signal-to-noise ratios," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22 (12), 2014.

[6] Z. Wang and D. Wang, "A joint training framework for robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24 (4), 2016.

[7] V. Peddinti, T. N. Sainath, S. Maymon, B. Ramabhadran, D. Nahamoo and V. Goel, "Deep Scattering Spectrum with deep neural networks," *in Proc. ICASSP*, 2014.

[8] T. N. Sainath, B. Kingsbury, A-r Mohamed and B. Ramabhadran, "Learning filter banks within a deep neural network framework," *in Proc. ASRU*, 2013.

[9] D. Palaz, R. Collobert, and M. Magimai.-Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks," *in Proc. Interspeech*, 2013.

[10] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNS," *in Proc. Interspeech*, 2015.

[11] Y. Hoshen, R. J. Hoshen, R. J. Weiss and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," *in Proc. ICASSP*, 2015.

[12] P. Ghahremani, V. Manohar, D. Povey and S. Khudanpur, "Acoustic modelling from the signal domain using CNNs," *in Proc. Interspeech*, 2016.

[13] O. Ichikawa, T. Fukuda and M. Nishimura, "Local peak enhancement combined with noise reduction algorithms for robust automatic speech recognition in automobiles," *in Proc. ICASSP*, 2008.

[14] T. Fukuda, O. Ichikawa and M. Nishimura, "Long-term spectro-temporal and static harmonic features for voice activity detection," *Selected Topics in Signal Processing, IEEE Journal of 4(5)*, pp. 816 - 823, IEEE, 2010.

[15] O. Ichikawa, S. J. Rennie, T. Fukuda and M. Nishimura, "Model-based noise reduction leveraging frequency-wise confidence metric for in-car speech recognition," *in Proc. ICASSP*, 2012.

[16] R. Hsiao, J. Ma, W. Hartmann, M. Karafi´at, F. Gr´ezl and et.al, "Robust speech recognition in unknown reverberant and noisy conditions," *in Proc. ASRU*, 2015.

[17] M. L. Seltzer and D. Yu and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," *in Proc. ICASSP*, 2013.

[18] S. Kim, B. Raj and I. Lane, "Environmental noise embeddings for robust speech recognition," *Computing Research Repository, arXiv,*1601.02553, 2016.

[19] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, A. Sehr, W. Kellermann, et al., "The Reverb Challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," *in Proc. WASPAA*, 2013.

[20] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," *in Proc. ICASSP*, 2013.

[21] G. Dahl, D. Yu, L. Deng and A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition," *IEEE Transactions on Audio Speech and Language Processing*, 1-1, 2010.

[22] M. Harper, "The Automatic Speech recogition In Reverberant Environments (ASpIRE) challenge," *in Proc. ASRU*, 2015.