INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Coping with Unseen Data Conditions: Investigating Neural Net Architectures, Robust Features, and Information Fusion for Robust Speech Recognition

Vikramjit Mitra, Horacio Franco

Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA
{vikramjit.mitra, horacio.franco}@sri.com

Abstract

The introduction of deep neural networks has significantly improved automatic speech recognition performance. For real-world use, automatic speech recognition systems must cope with varying background conditions and unseen acoustic data. This work investigates the performance of traditional deep neural networks under varying acoustic conditions and evaluates their performance with speech recorded under realistic background conditions that are mismatched with respect to the training data. We explore using robust acoustic features, articulatory features, and traditional baseline features against both in-domain microphone channel-matched and channel-mismatched conditions as well as out-of-domain data recorded using far- and near-microphone setups containing both background noise and reverberation distortions. We investigate feature-combination techniques, both outside and inside the neural network, and explore neural-network-level combination at the output decision level. Results from this study indicate that robust features can significantly improve deep neural network performance under mismatched, noisy conditions, and that using multiple features reduces speech recognition error rates. Further, we observed that fusing multiple feature sets at the convolutional layer feature-map level was more effective than performing fusion at the input feature level or at the neural-network output decision level.

Index Terms: automatic speech recognition, robust speech recognition, reverberation robustness, noise robustness, convolutional neural networks, feature fusion.

1. Introduction

Deep neural network (DNN)-based speech recognition systems [1-3] have demonstrated impressive performance in almost all evaluation conditions that have been reported so far. Unfortunately, DNN-hidden Markov model (HMM) systems are both quite data hungry and data sensitive [4]. These systems often show exceptional performance for in-domain data, but less so for out-of-domain (unseen or mismatched) data conditions. DNN models can be quite sensitive to data mismatch, and changes in the background acoustic conditions can result in catastrophic failure of such models. Further, any unseen distortion introduced at the input layers of the DNN results in a chain reaction of distortion propagation through the DNN.

Typically, as shown in this work, deeper neural nets can offer better speech recognition performance for in-domain data than shallower neural nets; while the shallower nets are relatively robust to unseen data conditions. This observation is a direct consequence of distortion propagation through the hidden layers of the neural nets, where typically deeper neural nets have more distorted information at their output activation level compared to the shallower ones, as shown by [5]. The literature reports that data augmentation, where data is collected from diverse sources [6] or artificially fabricated to match the evaluation condition [7, 8], improves the robustness of DNN acoustic models and combats data mismatch. In all such conditions, the assumption is that we have *a priori* knowledge about the kind of distortion that the model will see, which is often quite difficult, if not impossible, to anticipate. ASR systems deployed in the wild encounter unpredictable and highly dynamic acoustic conditions that are unique and hence difficult to augment.

A series of speech recognition challenges—such as MGB [9], CHiME-3 [10], ASpIRE [11], REVERB-2014 [12], and many more—has revealed the vulnerability of DNN systems to realistic acoustic conditions and variations, and has resulted in innovative ways for making DNN-based acoustic models more robust and less sensitive to unseen data conditions.

Typically, robust acoustic features are used to improve acoustic models when dealing with noisy and channel-degraded acoustic data [13, 14]. Recently, it was shown [6] that instead of performing ad-hoc signal processing as typically is done for robust feature generation, one can directly use the raw signal and employ a long short-term memory (LSTM) neural net to perform the signal processing for a DNN acoustic model where the parameters of the feature-extraction step are jointly optimized with the acoustic model parameters. Although such an approach is intriguing for future speech recognition research, unknown are both whether the limited training data would impact acoustic model behavior and how well such systems generalize to unseen data conditions, where feature transforms learned in a data-driven way may not generalize well for out-of-domain acoustic data.

Data adaptation is another alternative for dealing with unseen acoustic data. Several studies have explored novel ways of performing unsupervised adaptation of DNN acoustic models [15–17], where techniques based on maximum likelihood linear regression (MLLR) transforms, i-vectors, etc. have shown impressive performance gains over un-adapted models. Supervised adaptation with a limited set of transcribed target domain data is typically found to be helpful, and such approaches mostly involve updating the DNN parameters with the supervised adaptation data with some regularization. The effectiveness of such approaches is usually proportional to the volume of available adaptation data; however, such systems are typically found to digress away from the original training data and to learn the details of the target adaptation data. A solution to cope with this issue was proposed in [18], where a Kullback-Leibler divergence (KLD) regularization was proposed for DNN adaptation, which differs from the typically used L2 regularization [19] in the sense that it constrains the

http://dx.doi.org/10.21437/Interspeech.2016-966

model parameters themselves rather than the output probabilities.

In this work, we focus on investigating if robust features can provide an invariant representation such that they make the DNN acoustic models less prone to failure in the case of realistic and unseen acoustic data. We also investigate the role of the convolutional layer in convolutional neural nets in generating feature maps that are robust to acoustic degradations. Finally, we explore information fusion at different levels:

(a) Feature-space fusion: Two features are concatenated and fed to a single DNN/CNN.

(b) Feature-map fusion: Two convolutional layers are used to process each of the two features, and the ensuing feature maps after max-pooling are combined and then fed to a fully connected DNN. Here, we propose a fused-CNN (fCNN) architecture, as detailed in Section 3.

(c) Output-layer fusion: Two parallel CNNs are jointly trained that share a common output layer. Here, we propose a parallel CNN (pCNN) architecture, as detailed in Section 3.

2. Dataset and features

The acoustic models in this work were trained by using the multi-conditioned noise- and channel-degraded training data from the Aurora-4 noisy *Wall Street Journal* (WSJ0) corpus. Aurora-4 contains a total of six additive noise types with channel matched and mismatched conditions. It was created from the standard 5K WSJ0 database and includes 7180 training utterances of approximately 15-hours duration and 330 test utterances. In our experiments, we used the 16 kHz sampled data.

In Aurora-4, two training conditions were specified: (1) clean training, which is the full SI-84 WSJ training set without added noise; and (2) multi-condition training, with approximately half of the training data recorded by using one microphone, and the other half recorded by using a different microphone, with different types of added noise at different signal-to-noise ratios (SNRs). The Aurora-4 test data includes 14 test sets from two different channel conditions and six different added noises in addition to the clean condition. The SNR was randomly selected between 0 and 15 dB for different utterances. The six noise types used were: car, babble, restaurant, street, airport, and train station. The evaluation set consists of 5K words under two different channel conditions. The original audio data for test conditions 1–7 was recorded with a Sennheiser microphone, while test conditions 8–14 were recorded by using a second microphone randomly selected from a set of 18 different microphones (more details in [20]). The evaluation set was typically partitioned as follows:

Set A: Clean, matched-channel condition, test condition 1

Set B: Noisy, matched-channel condition, test conditions 2–7

Set C: Clean, mismatched-channel condition, test condition 8

Set D: Noisy, mismatched-channel condition, test conditions 9–14

To perform evaluation on unseen real-world data, we used the real evaluation data distributed with the REVERB-2014 [12] challenge. This real data is borrowed from the MC-WSJ-AV corpus [21], which consists of utterances recorded in a noisy and reverberant room. The real evaluation data (denoted as REV14 in this paper), sampled at 16 kHz, contained 372 utterances split equally between near- and far-field microphone conditions.

2.1. Features

The features investigated in this study range from baseline features: mel-filterbank (MFB) energies; gammatone filterbanks (GFBs); robust features (normalized modulation coefficients (NMC) [22], damped oscillator coefficients (DOC) [23], modulation of medium duration speech amplitudes (MMeDuSA) [24]); and bottleneck articulatory features from a CNN-based speech-inversion system. We used gammatone filterbank energies (GFBs) as the acoustic features for our experiments. The MFBs were extracted by using the Kaldi speech recognition toolkit [25], with 40 MFBs extracted. The GFBs were extracted by using SRI International's implementation, in which a time-domain gammatone filterbank containing 40 channels was used, where the filters were equally spaced on the equivalent rectangular bandwidth (ERB) scale. For all features, unless specifically mentioned, the analysis window was 25.6 ms, with a frame advance of 10 ms. For the GFBs, a 15^{th} power root was used for the nonlinear compression.

For the NMCs [22], the amplitude modulation (AM) signal was extracted from bandlimited speech signals by using the Teager's energy operator [26]. The powers of the AM signals were then root compressed by using the 15^{th} root, resulting in a 40-dimensional feature vector (more details in [22]).

In addition to the above features, we also trained a convolutional neural net (CNN) for speech inversion (a.k.a. acoustic-to-articulatory inversion) by using synthetic English data generated for words borrowed from the CMU English dictionary [27]. The data is similar to that described in [28]. The CNN had three hidden layers and used the NMCs as input features and vocal tract constriction variables (a.k.a. TVs) (more details regarding TVs are found in [28]) as targets. The second hidden layer had a bottleneck (BN) layer with 40 neurons. The BN activation from the CNN-based speech-inversion system was used as a candidate feature in our experiments reported in this paper.

3. Speech recognition system

We trained different acoustic models such as DNNs; CNNs; time-frequency convolutional nets (TFCNNs) [29]; hybrid convolutional neural net (HCNN); and other variants of CNNs. Among the other variants of CNN, we explored CNNs that can accept multiple streams of input features, where information fusion is performed either at the convolution-layer feature-map level or at the final-layer context-dependent (CD) state level.

Typically, CNNs give lower WERs compared to DNNs [14] when using filterbank features for ASR tasks, and the GFBs performed better or as well as the MFBs. To generate the alignments necessary for training the CNN system, a GMM-HMM model was used to produce the senone labels. Altogether, the GMM-HMM system produced 3125 context-dependent (CD) states for the Aurora-4 training data. The input features to the acoustic models were formed by using a context window of 15 frames (7 frames on either side of the current frame).

The acoustic models were trained by using cross-entropy on the alignments from the GMM-HMM system. For the CNN, 200 convolutional filters of size 8 were used in the convolutional layer, and the pooling size was set to 3 without

overlap. The subsequent fully connected network had four hidden layers, with 1024 nodes per hidden layer, and the output layer included as many nodes as the number of CD states for the given dataset. The networks were trained by using an initial four iterations with a constant learning rate of 0.008, followed by learning rate halving based on cross-validation error decrease. Training stopped when no further significant reduction in cross-validation error was noted or when cross-validation error started to increase. Backpropagation was performed by using stochastic gradient descent with a mini-batch of 256 training examples. For the DNN systems, we used five layers with 1024 neurons in each layer, with similar learning criteria as the CNNs.

The TFCNN architecture is similar to [29], where two parallel convolutional layers are used at the input, one performing convolution across time, and the other across the frequency axis of the input filterbank features. TFCNNs are found to give better performance [29] compared to their CNN counterparts. Here, we used 75 filters to perform time convolution, and 200 filters to perform frequency convolution. For time and frequency convolution, eight bands were used. A max-pooling over three samples was used for frequency convolution, while a max-pooling over five samples was used for time convolution. The feature maps after both the convolution operations were concatenated and then fed to a fully connected neural net, which had 1024 nodes and four hidden layers.

The HCNN is a modified deep neural network architecture specifically designed to jointly model the acoustic and the articulatory space. In an HCNN, two parallel neural networks are trained simultaneously. These two parallel neural networks model two things: (1) learning the acoustic space through a time-frequency convolutional net and (2) learning a temporal feature trajectory space through a time convolutional net. These two convolution layers had the same parameter specification as that used for the TFCNNs. The time-convolution layer contained 30 filters, followed by a max-pooling over five samples. The fully connected DNN layers were different in size, with 1024 neurons used for the TFCNN, and 512 neurons used for the time convolutional net. Note that both the parallel networks were jointly trained.

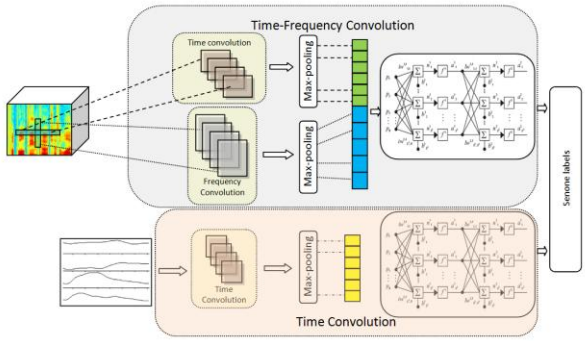

Figure 1: *Schematics of the hybrid convolutional neural network (HCNN). The top layer represents a TFCNN whose input is typically filterbank features, and the bottom layer represents the temporal features processed through time convolution.*

For information fusion, we investigated three approaches:

(1) Simple feature-fusion: a pair of acoustic feature was concatenated with each other, and the resultant was used to train a single DNN/CNN network.

(2) Feature-map-level fusion: two convolution layers were trained simultaneously for two different feature sets, and the ensuing feature maps were combined before training a fully connected network (fCNN), as shown in Figure 2.

(3) Decision-level fusion: two CNNs were jointly trained by sharing their output CD-state layer (pCNN), as shown in Figure 3.

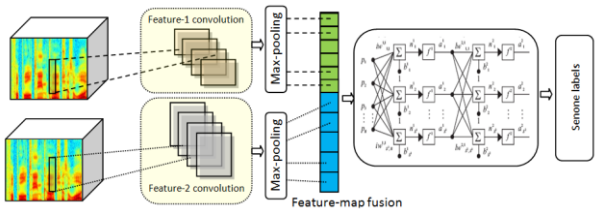

Figure 2: *CNN feature-map-level fusion* (*fCNN*).

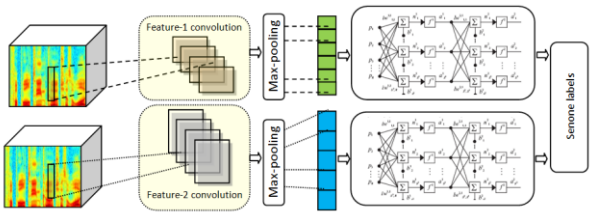

Figure 3: *CNN decision-level fusion* (*pCNN*).

4. Results

We first trained the individual feature-based DNN, CNN, and TFCNN acoustic models by using all the features described in Section 2. We report speech recognition performance in terms of word error rates (WERs). First, we compared performance using the baseline features. Table 1 shows the WERs for the MFB and GFB features for the Aurora-4 test conditions A, B, C and D, and the real reverberated evaluation data, REV14.

Table 1. *WER from DNN/CNN models trained with different features, when evaluated on Aurora-4 and REV14 test sets.*

	Features	Aurora-4					REV14
		A	B	C	D	avg.	real
DNN	MFB	3.5	7.1	8.0	18.9	12.0	75.8
	GFB	3.0	6.5	7.6	17.1	10.9	48.5
CNN	MFB	2.8	5.7	6.6	15.9	9.9	69.4
	GFB	2.6	5.6	5.4	14.2	9.1	46.7
	NMC	2.6	5.5	5.1	14.4	9.1	46.6
	DOC	2.7	5.6	5.2	14.2	9.0	49.4
	MMeDuSA	2.8	5.5	5.4	14.3	9.1	48.6

Table 1 demonstrates that GFB along with the robust features reduced the error rates compared to the MFB system. Although the MFB system was significantly impacted by the mismatched real REV14 data, the other features demonstrated reasonable performance. Note that the best single feature system for REV14 real evaluation data in [13] gave a WER of 30.4%, where the models were trained with reverberated data. In this experiment, the acoustic model was not trained with any reverberated condition; hence, a significant mismatch existed between the training and REV14 evaluation conditions, which resulted in a 50% increase in the error rates. Next, we investigated how the number of hidden layers in a

DNN affects the WER for the matched- and mismatched-condition evaluation sets, for which we selected the GFB features and trained DNNs with fewer hidden layers. Table 2 shows the WER for DNNs having three, four, and five hidden layers, trained with GFB features.

Table 2. *WER from DNN models having different number of hidden layers with 1024 neurons trained with GFB features.*

Feature	#Layers	Aurora-4					REV14
		A	B	C	D	avg.	real
GFB	5	3.0	6.5	7.6	17.1	10.9	48.5
GFB	4	3.0	6.6	7.2	16.7	10.7	48.2
GFB	3	3.3	6.8	7.5	17.0	11.0	47.4

Table 2 shows that the number of hidden layers in a DNN may determine how well the acoustic model would perform in unseen conditions. Note that both the C and D conditions of Aurora-4 were mismatched with respect to the training set in terms of the channel conditions. With reduction in the number of hidden layers, we observe that the WER for the matched conditions (i.e., Aurora-4 test conditions A and B) increases, but the WER decreases for the REV14 evaluation set and also for the Aurora-4 C and D evaluation sets. This finding may indicate that deeper DNNs may learn fine-grained information about the training data that may cripple them from coping with unseen acoustic conditions. Hence, for mismatched conditions, models with fewer layers may generalize better than models with more hidden layers.

Next, we explored TFCNNs and using articulatory features through HCNNs. In the case of the HCNNs, GFB was used as the acoustic feature, which was used with the BN features obtained from the CNN-based speech-inversion system. Table 3 presents the results from the TFCNNs and HCNNs, and shows that using articulatory BN features not only gave better results in Aurora-4 noisy evaluation conditions but also lower error rates in real REV14 conditions. Similar to our observation in [29], the TFCNNs gave lower WERs for almost all conditions compared to the CNNs.

Table 3. *WER from TFCNN and HCNN models.*

	Features	Aurora-4					REV14
		A	B	C	D	avg.	real
HCNN	GFB+BN articulatory features	2.9	5.5	5.2	14.0	8.9	46.0
TFCNN	GFB	2.7	5.6	5.4	14.0	9.0	45.9
	NMC	2.7	5.5	5.4	14.1	9.0	45.9
	DOC	2.8	5.4	5.1	14.1	8.9	48.4
	MMeDuSA	2.6	5.7	5.4	14.4	9.2	48.2

Next, we explored using multiple features, investigating simple feature fusion, feature-map fusion, and CNN-decision-level fusion for different pairs of features. As is evident from Tables 1 and 3, the GFB and NMC features demonstrated better performance, and we observed similar results from the feature-fusion experiments with models trained with these two features. Table 4 presents the WERs from DNN, CNN, pCNN, and fCNN systems trained with both GFB and NMC features. Note that for the DNN and CNN systems, the models were fed with a combined feature set as input.

Table 4. *WER from different fusion experiments using DNN, CNN, pCNN, and fCNN models.*

Systems	Aurora-4					REV14
	A	B	C	D	avg.	real
DNN	3.3	7.0	7.2	17.5	11.3	49.2
CNN	2.6	5.4	5.0	14.0	8.9	46.7
fCNN	2.7	5.4	5.4	13.6	8.7	46.7
pCNN	2.7	5.6	5.4	14.4	9.1	46.7

Table 4 shows that feature-map-level fusion was more effective than decision-level fusion, as a consequence the fCNN system trained with GFB and NMC features gave the lowest WER in our experiments. However, for unseen conditions, the best WER came from the TFCNN system reported in Table 3. The TFCNNs were originally proposed for combating reverberation effects, in which time convolution is meant to mitigate the temporal artifacts introduced by reverberation. As no reverberated data was used during training here, the learned time convolution layer was not as effective as reported in [29], but it was still a better-performing model than the CNNs.

5. Conclusions

In this work, we demonstrated how a DNN/CNN system performs when exposed to unseen data. For this purpose, we trained our acoustic model with noisy data and evaluated it with channel-degraded noisy data from Aurora-4 and real-world reverberated data from the Reverb-2014 evaluation set. The models were not trained with reverberation and, as a consequence, demonstrated an almost 50% increase in WER with respect to a similar model (reported in [13]) trained with reverberated data. We also demonstrated that networks with fewer hidden layers may generalize well for unseen data conditions, with a compromise of reduced performance on seen data conditions. We also found that feature-map-level fusion is an elegant strategy for fusing multiple feature streams and better for this purpose than using a simple feature combination or fusing multiple neural nets (individually trained with those features) at the output level.

In the future, we plan to investigate learning a low-dimensional representation through bottleneck layers and using them to fuse multiple feature streams. We will also investigate unsupervised adaptation approaches to adapt to unseen and realistic acoustic conditions.

6. Acknowledgements

This material is based upon work partly supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-15-C-0037. The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

7. References

[1] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. on ASLP*, vol. 20, no. 1, pp. 14 –22, 2012.

[2] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," *Proc. of Interspeech*, 2011.

[3] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kinsgbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, 2012.

[4] F. Grézl, E. Egorova and M. Karafiát, "Further investigation into multilingual training and adaptation of stacked bottle-neck neural network structure," *Proc. of SLT*, pp. 48–53, 2014.

[5] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," *Proc. of JMLR: Workshop and Conference Proceedings, Workshop on Unsupervised and Transfer Learning*, pp. 1–20, 2011.

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

[7] V. Peddinti, G. Chen, V. Manohar, T. Ko, D. Povey and S. Khudanpur, "JHU ASpIRE system: Robust LVCSR with TDNNs, i-vector adaptation and RNN-LMS," *Proc. of ASRU*, 2015.

[8] M. Karafiát, F. Grézl, L. Burget, I. Szöke and J. Cernocký "Three ways to adapt a CTS recognizer to unseen reverberated speech in BUT system for the ASpIRE challenge," *Proc. of Interspeech*, pp. 2454–2458, 2015.

[9] P. Bell, M. J. F. Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester and P.C. Woodland, "The MGB challenge: Evaluating multi-genre broadcast media recognition," *Proc. of ASRU*, 2015.

[10] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," *Proc. of ASRU*, 2015.

[11] M. Harper, "The automatic speech recognition in reverberant environments (ASpIRE) challenge," *Proc. of ASRU*, 2015.

[12] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot and B. Raj, "The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013.

[13] V. Mitra, W. Wang and H. Franco, "Deep convolutional nets and robust features for reverberation-robust speech recognition," in *Proc. of SLT*, pp. 548–553, 2014.

[14] V. Mitra, W. Wang, H. Franco, Y. Lei, C. Bartels and M. Graciarena, "Evaluating robust features on deep neural networks for speech recognition in noisy and channel mismatched conditions," in *Proc. of Interspeech*, pp. 895–899, Singapore, 2014.

[15] T. Yoshioka, A. Ragni, M. J. F. Gales, "Investigation of unsupervised adaptation of DNN acoustic models with filterbank input," *Proc. of ICASSP*, pp. 6344–6348, 2014.

[16] G. Saon, H. Soltau, D. Nahamoo and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," *Proc. of ASRU*, pp. 55–59, 2013.

[17] S.H.K. Parthasarathi, B. Hoffmeister, S. Matsoukas, A. Mandal, N. Strom and S. Garimella, "fMLLR based feature-space speaker adaptation of DNN acoustic models," *Proc. of Interspeech*, 2015.

[18] D. Yu, K. Yao, H. Su, G. Li and F. Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," *Proc. of ICASSP*, 2013.

[19] X. Li and J. Bilmes, "Regularized adaptation of discriminative classifiers," *Proc. ICASSP'06*, 2006.

[20] G. Hirsch, "Experimental framework for the performance evaluation of speech recognition front-ends on a large vocabulary task," *ETSI STQ-Aurora DSR Working Group*, June 4, 2001.

[21] M. Lincoln, I. McCowan, J. Vepa and H. K. Maganti, "The multi-channel *Wall Street Journal* audio visual corpus (MC-WSJ-AV): Specification and initial experiments," *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding*, 2005.

[22] V. Mitra, H. Franco, M. Graciarena and A. Mandal, "Normalized amplitude modulation features for large vocabulary noise-robust speech recognition," *Proc. of ICASSP*, pp. 4117–4120, Japan, 2012.

[23] V. Mitra, H. Franco and M. Graciarena, "Damped oscillator cepstral coefficients for robust speech recognition," *Proc. of Interspeech*, pp. 886–890, Lyon, 2013.

[24] V. Mitra, H. Franco, M. Graciarena and D. Vergyri, "Medium duration modulation cepstral feature for robust speech recognition," *Proc. of ICASSP*, pp. 1768–1772, Florence, 2014.

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz et al., "The Kaldi speech recognition toolkit," *Proc. ASRU*, 2011.

[26] H. Teager, "Some observations on oral air flow during phonation," *IEEE Trans. ASSP*, pp. 599–601, 1980.

[27] http://www.speech.cs.cmu.edu/cgi-bin/cmudict.

[28] V. Mitra, G. Sivaraman, H. Nam, C. Espy-Wilson and E. Saltzman, "Articulatory features from deep neural networks and their role in speech recognition," *Proc. of ICASSP*, pp. 3041–3045, Florence, 2014.

[29] V. Mitra and H. Franco, "Time-frequency convolution networks for robust speech recognition," *Proc. of ASRU*, 2015.