INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Improved Automatic Speech Recognition using Subband Temporal Envelope Features and Time-delay Neural Network Denoising Autoencoder

Cong-Thanh Do and Yannis Stylianou

Toshiba Research Europe, 208 Cambridge Science Park, Cambridge, United Kingdom
{cong-thanh.do, yannis.stylianou}@crl.toshiba.co.uk

Abstract

This paper investigates the use of perceptually-motivated subband temporal envelope (STE) features and time-delay neural network (TDNN) denoising autoencoder (DAE) to improve deep neural network (DNN)-based automatic speech recognition (ASR). STEs are estimated by full-wave rectification and low-pass filtering of band-passed speech using a Gammatone filter-bank. TDNNs are used either as DAE or acoustic models. ASR experiments are performed on Aurora-4 corpus. STE features provide 2.2% and 3.7% relative word error rate (WER) reduction compared to conventional log-mel filter-bank (FBANK) features when used in ASR systems using DNN and TDNN as acoustic models, respectively. Features enhanced by TDNN DAE are better recognized with ASR system using DNN acoustic models than using TDNN acoustic models. Improved ASR performance is obtained when features enhanced by TDNN DAE are used in ASR system using DNN acoustic models. In this scenario, using STE features provides 9.8% relative WER reduction compared to when using FBANK features.

Index Terms: Subband temporal envelope, denoising autoencoder, time-delay neural network, deep neural network, acoustic features

1. Introduction

Feature extraction is an active research topic in automatic speech recognition (ASR) research. To date, log-mel filter-bank (FBANK) features [1], which are created by skipping the discrete cosine transform (DCT) in the Mel frequency cepstral coefficients (MFCCs) [2] computation, are among the most popular features for deep neural network (DNN)-based ASR.

Energy peaks in frequency bands of speech signal reflect the resonant properties of the vocal tract and thus provide acoustic information on the production of speech sound [3]. Acoustic features for ASR, such as MFCCs or perceptual linear predictive (PLP) coefficients [4], attempt to capture these resonant properties from speech signal in spectral domain. Similarly, these information can be extracted in the time domain from the subband temporal envelopes (STEs). STEs are temporal envelopes of subband signals resulting from band-pass filtering of the original speech signal. Time-domain feature extraction could be thus as effective as spectral domain feature extraction.

Temporal information has been included in acoustic features for ASR, for instant in TRAPS (TempoRAl Patterns) features [5]. These features carry subband temporal information and are extracted in the spectral domain. STEs of speech signal have been studied in speech intelligibility [6] or in speech signal processing for cochlear implants [7]. In addition, STEs have been studied in speech signal processing and acoustic feature extraction for ASR, based on hidden Markov models (HMMs) [8, 9, 10, 11]. In [10, 11], acoustic features are computed from the power of the short- (25 ms) or medium-term ($\sim$ 200 ms) amplitude modulations, and then, transform to cepstral domain using a discrete cosine transform (DCT).

In this paper, we investigate the use of STE features and time-delay neural network (TDNN) denoising autoencoder (DAE) to improve DNN-based ASR performance. STE features can be used in ASR systems using DNN or TDNN acoustic models (AMs) or as input to TDNN DAE for feature enhancement. Performance of ASR systems using STE features is compared with that of ASR systems using conventional FBANK features. Experimental results show that improved ASR performance is observed when features enhanced by TDNN DAE are used in ASR system using DNN AMs. Systems using STE features have better performance than when using FBANK features. The paper is organized as follows. Section 2 mentions relevant works. STE features are presented in section 3. TDNN DAE is presented in section 4. Section 5 presents ASR experimental results on Aurora-4 corpus [12]. Conclusions are presented in section 6.

2. Relation to prior work

Feature enhancement is one of the approaches for improving ASR performance. It aims at enhancing the acoustic features extracted from noisy input speech and use the enhanced features for recognition. Feature enhancement using neural networks has been investigated in the literature [13, 14, 15, 16].

A neural network which attempts to estimate a cleaner version of its noisy input features is known in the literature as a DAE [17]. It is known that neural network architectures used for feature enhancement often take into account long temporal context. In [13], recurrent neural network (RNN), and in [14, 18], bidirectional long-short term memory (LSTM) neural networks have been used as DAE, respectively.

The common characteristic that makes RNN and BLSTM architecture relevant for DAE being their ability to model the temporal evolution of speech and noise over a longer period of time [13, 14]. TDNN was shown to be able to learn long-term temporal contexts from speech signal [19]. This ability suggests that TDNN could be effectively used as DAE.

STE features track energy peaks in perceptual frequency bands which reflect the resonant properties of the vocal tract. These are temporal information about transients that is not present in conventional features, such as MFCCs, PLP coefficients or FBANK features [20]. Temporal context information from speech could be thus better extracted by STE features. In addition, STE features in the present paper are extracted from slowly-varying temporal envelopes; this could help reducing redundancy more efficiently. These characteristics suggest that STE features could provide additional benefits when being used with DAE based on neural network architectures which learn long-term temporal contexts, for instance the TDNN DAE. This utilization will be investigated in this paper.

http://dx.doi.org/10.21437/Interspeech.2017-1096

3. Subband temporal envelope features

Given a speech signal $s(n)$, the M STE signals $e_m(n)$, $m = 1, \ldots, M$ of $s(n)$ are extracted as follows. The speech signal $s(n)$ is first pre-emphasized by using a filter having a transfer function $H(z) = 1 - 0.97z^{-1}$. The pre-emphasized speech signal is then decomposed into M subband signals $s_m(n)$, $k = 1, \ldots, M$ using a filter-bank consisting of M Gammatone band-pass filters. In this work, Gammatone filters implementation from [21] is used. Each Gammatone band-pass filter in the filter-bank is implemented as a cascade of four separate second order IIR (infinite impulse response) filters. This implementation is done to avoid round-off errors [21]. The frequency responses of the Gammatone filters are shown in Fig. 1. The center frequencies of the Gammatone filter are linearly spaced on the ERB (equivalent rectangular bandwidth) scale with the first one starts at 100 Hz.

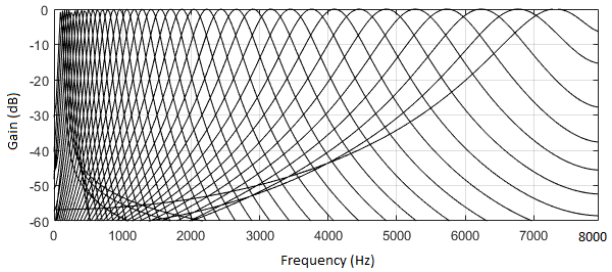

Figure 1: *Frequency responses of the Gammatone filters in the analysis filter-bank.*

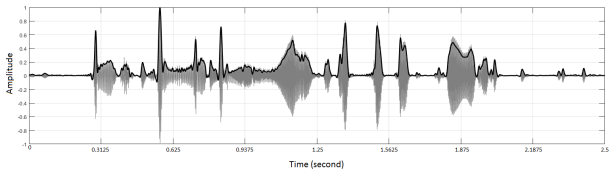

Figure 2: *Subband temporal envelope (STE) (bold line) extracted from the subband signal resulting from the 8th Gammatone band-pass filter in the analysis filter-bank. The low-pass filter for extracting STE has a cut-off frequency of 50 Hz.*

STEs $e_m(n)$, $m = 1, \ldots, M$ of the subband signals $s_m(n)$, $m = 1, \ldots, M$ are then extracted by, first, full-wave rectifying the subband signals followed by low-pass filtering of the resulting signals. In this work, the low-pass filter for extracting STEs is a fourth-order elliptic low-pass filter with 2-dB of peak-to-peak ripple and a minimum stop-band attenuation of 50-dB. The cut-off frequency of this low-pass filter, which controls the bandwidth of STEs, is 50 Hz because this cut-off frequency ensures a reasonable STE bandwidth for human and machine speech recognition [3, 8, 9]. An example of STE, extracted by this method, is shown in Fig. 2.

The STE feature extraction is shown in Fig. 3. From STEs $e_m(n)$, $m = 1, \ldots, M$ extracted from the whole utterance, short-term frames of 25 ms are extracted every 10 ms. The short-term frames are multiplied with Hamming windows to emphasize the samples in the middle of the analysis frames. At time instant k, assume that $\hat{e}_{m,k}(n), m = 1, \ldots, M$ are the short-term STEs obtained after the Hamming windowing, a feature vector $\mathbf{y}_k = [y_{1,k}, y_{2,k}, \ldots, y_{M,k}]^T$ is extracted. A feature coefficient $y_{m,k}$ is computed as:

$$y_{m,k} = \frac{1}{N} \sum_{n=1}^{N} \hat{e}^2_{m,k}(n) \quad (1)$$

where N is number of samples in a frame and k is frame index. Finally, these feature coefficients are root compressed with $1/15^{th}$ root, according to a compression suggested in [11].

The way STE features are estimated in this paper has two major differences compared to previously proposed acoustic features based on temporal envelopes [10, 11]. First, STE features in this paper are proposed for DNN-based ASR. It is known that DNNs do not require uncorrelated data [1] as Gaussian mixture models (GMMs) do in acoustic modeling. Therefore, there is no decorrelation within the proposed STE feature extraction whereas DCT was used in [10, 11] to decorrelate feature coefficients. Second, STE feature extraction in this paper uses full-wave rectification and low-pass filtering [3, 8, 9] to extract STEs. In [10], frequency domain linear prediction (FDLP) and in [11], discrete energy separation algorithm (DESA) were used to extract temporal envelopes, respectively. By using full-wave rectification and low-pass filtering to extract STEs, bandwidth of STEs as well as their temporal resolution can be controlled by the cut-off frequency of the low-pass filter.

4. Time-delay neural network denoising autoencoder

TDNN architecture was introduced in [19]. This neural network architecture can represent relationships between events in time which could be spectral coefficients, but might also be the output of higher level feature detectors. In a TDNN architecture, the initial transforms are learned on narrower contexts and the deeper layers process the hidden activations from a wider temporal context. Hence the higher layers have the ability to learn wider temporal relationships [22]. In this architecture, the time delay of speech frames are explicitly modeled thanks to the delay units. Hence, temporal contexts are better modeled.

Back-propagation learning algorithm [23] is used to train TDNN DAE given input features extracted from noisy training speech and output features extracted from clean training speech. The back-propagation algorithm adjusts the TDNN's link weights to realize the feature enhancement mapping. The cost function that is used by the back-propagation algorithm is a square error measure between referenced clean output features and the actual TDNN's output. The actual TDNN's output is computed from noisy input features using actual TDNN weights. On every presentation of learning samples, each weight is updated in an attempt to decrease this square error measure. Details of the TDNN training is presented in section 5.2.1.

5. ASR experiments

5.1. Acoustic feature extraction

Performance of ASR systems using STE features is compared with that of ASR systems using conventional FBANK features. In this work, the FBANK features are extracted in a conventional manner as follows: speech signal is first pre-emphasized by using a filter having a transfer function $H(z) = 1-0.97z^{-1}$. Speech frames of 25 ms are then extracted every 10 ms and multiplied with Hamming windows. Discrete Fourier transform (DFT) is used to transform speech frames into spectral domain. Sums of the element-wise multiplication between magnitude spectrum and Mel-scale filter-bank are computed. The FBANK

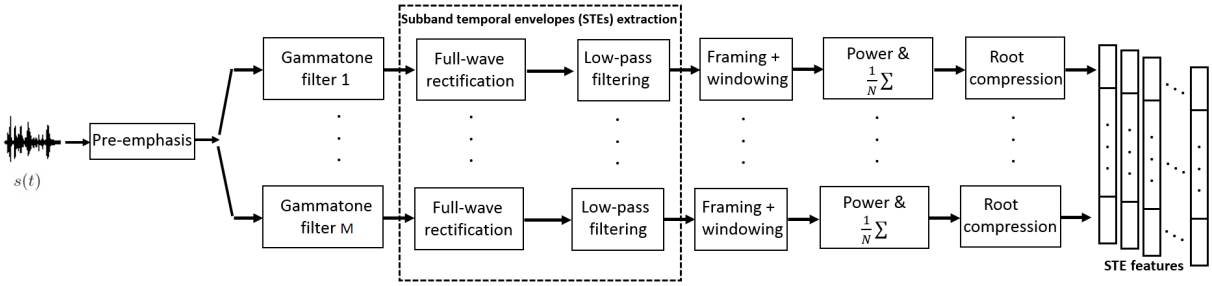

Figure 3: *Algorithm for STE feature extraction.* $\frac{1}{N}\sum$ *means computing the average over a speech frame (see equation 1).*

coefficients are obtained by taking logarithm of these sums. 40-channel Mel-scale filter-bank is used and 1 frame's total energy coefficient is appended, resulting in 41-dimensional static FBANK features. These FBANK features are extracted using HTK toolkit [24].

STE features are extracted as in section 3. In STE feature extraction, 40 band-pass filters are used in the Gammatone filter-bank. The frame's total energy is used as an additional coefficient. The dimension of the static STE features is thus 41, same as the FBANK features. In this work, we use a cut-off frequency of 50 Hz to extract STEs. This cut-off frequency ensures a reasonable STE bandwidth for human and machine speech recognition [3, 8, 9].

5.2. Speech recognition systems training

ASR systems are trained and evaluated using Aurora-4 corpus [12]. Aurora-4 is a medium vocabulary task based on the Wall Street Journal (WSJ0) corpus. There are two training sets: clean and multi-condition. Each set consists of 7137 utterances from 83 speakers. All the utterances in the clean training set were recorded by the primary Sennheiser microphone which is close-talking microphone. The multi-condition training set was created by keeping half of the clean training set and replacing other half by same speech utterances which are simultaneously recorded by one of a number of different secondary microphones. Seventy five percent of the utterances in each half were corrupted by six different noises (airport, babble, car, restaurant, street, and train) at 10-20 dB signal to noise ratio (SNR).

The evaluation set was derived from WSJ0 5K-word closed-vocabulary test set which consists of 330 utterances spoken by 8 speakers. This test set was recorded by the primary microphone and a secondary microphone. 14 test sets were created by corrupting these two sets by the same six noises used in the training set at 5-15 dB SNR. The types of noises are matched across training and test sets but the SNRs of the data are partially mismatched. These 14 test sets can be grouped into 4 subsets: clean, noisy, clean with channel distortion, noisy with channel distortion, which will be referred to as A, B, C, and D, respectively [25]. All the data used for the experiments in this paper are sampled at 16 kHz.

5.2.1. TDNN DAE training

Acoustic features are extracted from both clean and multi-condition training data of Aurora-4 to train TDNN DAE. A summary of the TDNN DAE training is shown in Fig. 4(a). The TDNN DAE training is done using back-propagation based on square error criterion. In [22], a TDNN architecture was proposed for acoustic modeling which allows flexible selection of input contexts of each layers required to compute an output activation, at one time step. A major difference between the architecture proposed in [22] and the original architecture proposed in [19] is the use of p-norm non-linearity which is a dimension reducing non-linearity [26].

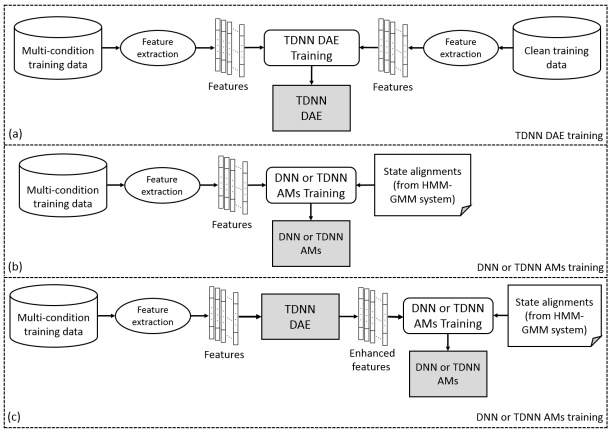

Figure 4: *Training of TDNN DAE (Fig. 4(a)), DNN and TDNN AMs (Figs. 4(b) and 4(c)).*

In the present paper, we use the TDNN architecture proposed in [22] for DAE. The TDNN architecture uses group size of 10, and 2-norm. In all hidden layers the p-norm input and output dimensions are 3000 and 300, respectively. The input contexts of each layer required to compute an output activation define the TDNN architecture. In this architecture, asymmetric input contexts, with more context to the left, seemed to work better. We thus apply a 4 hidden layers TDNN architecture with an input temporal context of [t-13, t+9] which was found to be optimal in [22]. In this configuration, the highest hidden layer covers a context of 13 frames on the left and 9 frames on the right of the current frame. The layer-wise context of this architecture are [-2,2], {-1,2}, {-3,4}, {-7,2}, {0}, the same as in the reported optimal architecture. For the sake of reference, we use similar notations as in [22] to describe layer-wise contexts. According to this, [-2,2] means 5 frames at offsets -2, -1, 0, 1, 2 compared to the current frame are spliced together for the computation of hidden activations in the first hidden layer. Assume that t_1 and t_2 are two positive integers, {-t_1,t_2} means two frames at offsets $-t_1$ and t_2 compared to the current frame are spliced to compute hidden activations in the corresponding hidden layer. {0} is a conventional, non-splicing hidden layer. This sub-sampling scheme was proposed in [22] to reduce training time and model size.

5.2.2. TDNN and DNN AMs training

The 4 hidden layers TDNN architecture in section 5.2.1 is also used to train TDNN AMs. The state alignments for TDNN AMs training are obtained from a speaker adaptive training (SAT) HMM-GMM system, trained on the multi-condition training data using MFCCs features. These are the state alignments used for training all AMs, including DNN AMs, in the present paper. TDNN AMs training are done using back-propagation based on cross-entropy criterion [22]. DNN AMs are trained according to the standard training recipe in [27] using Kaldi speech recognition toolkit [28]. TDNN and DNN AMs are trained by acoustic features extracted from multi-condition training data of Aurora-4. Features extracted from multi-condition training data can be enhanced by the TDNN DAE before being used for training TDNN or DNN AMs. These training are summarized in Fig. 4(b) and 4(c). In the DNN AMs training, a conventional context window of 11 frames is used. In HMM-GMM, HMM-TDNN and HMM-DNN systems training, utterance-level mean normalization is performed on static features. First and second-order delta features are then appended. The DNNs are initialized using layer-by-layer generative pre-training and then discriminatively trained using back-propagation based on cross-entropy criterion.

In the present work, the DNNs consist of 7 hidden layers, each layer has 2048 nodes. This DNN architecture was found to be appropriate for Aurora-4 corpus [25]. The activation function is sigmoid. The softmax layer consists of 2298 nodes which are the number of tied context-dependent acoustic states. Decoding is performed with the task-standard WSJ0 bi-gram language model.

5.3. Experimental results

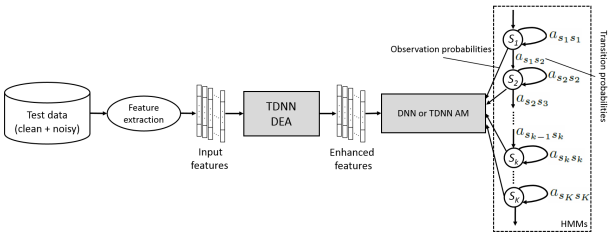

Figure 5: *Enhancing features with TDNN DAE prior to recognition by ASR system using DNN or TDNN AMs.*

ASR systems for experiments use either DNN or TDNN AMs. TDNN DAE is used to enhance features prior to recognition with these systems (see Fig. 5). Experimental results, in terms of word error rates (WERs), are shown in Tables 1 and 2. Table 1 shows the performance of conventional FBANK and STE features in ASR systems using DNN or TDNN AMs. STE features provide 2.2% and 3.7% relative WER reduction compared to FBANK features when DNN and TDNN are used as AMs respectively. Lattices of ASR systems using FBANK and STE features are combined using minimum Bayes risk decoding [29]. In these combinations, equal weights are assigned to each system. When two individual systems use DNN AMs, a relative gain of 5.2% WER compared to baseline system using FBANK features is obtained with the combined system. The corresponding relative gain obtained when two individual systems used TDNN AMs is 5.9% WER.

Table 1: *Performance (WERs) of FBANK and STE features in ASR systems using DNN or TDNN AMs. Results of system combination are also presented.*

Features \ Condition	A	B	C	D	Avg.
FBANK (DNN AM)	4.1	8.2	9.9	21.2	13.6
STE (DNN AM)	3.9	8.1	10.4	20.6	13.3
System combination	3.9	7.7	9.5	20.0	12.9
FBANK (TDNN AM)	4.3	8.3	9.0	21.2	13.6
STE (TDNN AM)	3.9	8.1	9.0	20.4	13.1
System combination	3.9	7.8	8.4	20.0	12.8

Table 2: *Performance (WERs) of FBANK and STE features in ASR systems where TDNN DAE is used prior to recognition with systems using DNN or TDNN AMs.*

Features \ Condition	A	B	C	D	Avg.
FBANK (DAE+DNN AM)	3.9	8.0	8.2	20.9	13.3
STE (DAE+DNN AM)	3.9	7.1	7.1	18.9	12.0
System combination	3.7	6.8	7.1	18.2	11.5
FBANK (DAE+TDNN AM)	4.3	8.3	8.6	21.8	13.8
STE (DAE+TDNN AM)	4.1	7.9	7.7	20.4	13.0
System combination	3.7	7.3	7.3	20.0	12.5

Table 2 shows performance of ASR systems in which TDNN DAE is used to enhance features prior to recognition using DNN or TDNN AMs. When TDNN DAE is used prior to systems using DNN AMs, improved performance is observed. Relative gains of 2.2% WER and 9.8% WER compared to system using only DNN AMs are obtained with TDNN DAE + DNN AM systems, when FBANK and STE features are used respectively (see Table 1). In this scenario, using STE features provides a relative gain of 9.8% WER compared to when using FBANK features. This result shows that TDNN DAE outputs better features for ASR when STE features are provided as input. The effectiveness of TDNN DAE when used with STE features could be made thanks to better context information extracted by STE features. In addition, STE features are extracted from slowly-varying temporal envelopes; this could help reducing redundancy more efficiently.

Combining two TDNN DAE + DNN AM systems which use FBANK and STE features provides a relative gain of 13.5% WER compared to baseline TDNN DAE + DNN AM system using FBANK features. However, when TDNN DAE is used prior to systems using TDNN AMs, no performance gain is observed even though system combination still provides improvement. Indeed, TDNN AMs which compute hidden activations from a limited number of nodes might not be able to exploit useful information from the output of TDNN DAE as efficiently as the fully-connected layers in DNN AMs do.

6. Conclusion

The paper investigated the use of STE features and TDNN DAE to improve DNN-based ASR. Features enhanced by TDNN DAE were better recognized with ASR systems using DNN AMs than using TDNN AMs. Improved ASR performance was observed when features enhanced by TDNN DAE were used in ASR system using DNN AMs. In this scenario, using STE features provided a relative gain of 9.8% WER compared to when using conventional FBANK features. System combination further improved the ASR performance.

7. References

[1] A.-R. Mohamed, G. Hinton, and G. Penn, "Understanding how deep belief networks perform acoustic modelling," in *Proc. IEEE ICASSP*, Kyoto, Japan, March 2012, pp. 4273–4276.

[2] S. B. Davis and P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[3] R. V. Shannon, F.-G. Zeng, V. Kamath, J. Wygonski, and M. Ekelid, "Speech recognition with primarily temporal cues," *Science*, vol. 207, no. 5234, pp. 303–304, 1995.

[4] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *J. Acoust. Soc. Am.*, vol. 87, no. 4, pp. 1738–1752, 1990.

[5] H. Hermansky and S. Sharma, "Temporal patterns (TRAPs) in ASR of noisy speech," in *Proc. IEEE ICASSP*, Phoenix, USA, March 1999, pp. 289–292.

[6] F. Apoux, S. E. Yoho, C. L. Youngdahl, and E. W. Healy, "Role and relative contribution of temporal envelope and fine structure cues in sentence recognition by normal-hearing listeners," *J. Acoust. Soc. Am.*, vol. 134, pp. 2205–2212, 2013.

[7] P. C. Loizou, "Mimicking the human ear: an overview of signal processing strategies for converting sound into electrical signals in cochlear implants," *IEEE Signal Processing Magazines*, vol. 15, no. 5, pp. 101–130, 1998.

[8] C.-T. Do, *Subband temporal envelopes of speech signal and their central role in robust ASR and perceptually-motivated speech signal processing*. PhD Thesis, Telecom Bretagne, 2010.

[9] C.-T. Do, D. Pastor, and A. Goalic, "A novel framework for robust ASR using cochlear implant-like spectrally reduced speech," *Speech Communication*, vol. 54, pp. 119–133, 2012.

[10] S. Thomas, S. Ganapathy, and H. Hermansky, "Phoneme recognition using spectral envelope and modulation frequency features," in *Proc. IEEE ICASSP*, Taiwan, April 2009, pp. 4453–4459.

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

[12] N. Parihar and J. Picone, *Aurora working group: DSR front end LVCSR evaluation: AU/384/02*. Institute for Signal and Information Processing Technical Report, 2002.

[13] A. L. Maas, V. Q. Le, T. M. ONeil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Proc. ISCA INTERSPEECH*, Portland, OR, USA, September 2012, pp. 22–25.

[14] M. Wollmer, Z. Zhang, F. Weninger, B. Schuller, and G. Rigoll, "Feature enhancement by bidirectional LSTM networks for conversational speech recognition in highly non-stationary noise," in *Proc. IEEE ICASSP*, Vancouver, Canada, May 2013, pp. 6822–6826.

[15] X. Feng, Y. Zhang, and J. Glass, "Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition," in *Proc. IEEE ICASSP*, Florence, Italy, May 2014, pp. 1778–1782.

[16] K. Han, Y. He, D. Bagchi, E. Fosler-Lussier, and D. Wang, "Deep neural network based spectral feature mapping for robust speech recognition," in *Proc. ISCA INTERSPEECH*, Dresden, Germany, September 2015, pp. 2484–2488.

[17] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. International Conference on Machine Learning*, Helsinki, Finland, July 2008, pp. 1096–1103.

[18] F. Weninger, S. Watanabe, Y. Tachioka, and B. Schuller, "Deep recurrent de-noising auto-encoder and blind de-reverberation for reverberated speech recognition," in *Proc. IEEE ICASSP*, Florence, Italy, May 2014, pp. 4656–4660.

[19] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, 1989.

[20] N. Morgan, Q. Zhu, A. Stolcke, K. Sonmez, S. Sivadas, T. Shinozaki, M. Ostendorf, P. Jain, H. Hermansky, D. Ellis, G. Doddington, B. Chen, O. Cetin, H. Bourlard, and M. Athineos, "Pushing the envelope - aside," *IEEE Signal Processing Magazine*, vol. 22, no. 5, pp. 81–88, 2005.

[21] M. Slaney, *Auditory Toolbox (version 2)*. Interval Research Corporation Technical Report #1998-010, 1998.

[22] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. ISCA INTERSPEECH*, Dresden, Germany, September 2015, pp. 3214–3218.

[23] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 9, pp. 533–536, 1986.

[24] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. L. G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book, version 3.4.1*. Cambridge University Engineering Department, 2006.

[25] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. IEEE ICASSP*, Vancouver, Canada, May 2013, pp. 7398–7402.

[26] X. Zhang, J. Trman, D. Povey, and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in *Proc. IEEE ICASSP*, Florence, Italy, May 2014, pp. 215–219.

[27] 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*, vol. 20, no. 1, pp. 30–42, 2012.

[28] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *Proc. IEEE ASRU 2011*, Hawaii, USA, December 2011.

[29] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum Bayes risk decoding and system combination based on a recursion for edit distance," *Computer Speech and Language*, vol. 25, no. 4, pp. 802–828, 2011.