RECURRENT DEEP STACKING NETWORKS FOR SUPERVISED SPEECH SEPARATION

Zhong-Qiu Wang♣ *and DeLiang Wang*♣, ♥

♣Department of Computer Science and Engineering, The Ohio State University, USA
♥Center for Cognitive and Brain Sciences, The Ohio State University, USA
{wangzhon, dwang}@cse.ohio-state.edu

ABSTRACT

Supervised speech separation algorithms seldom utilize output patterns. This study proposes a novel recurrent deep stacking approach for time-frequency masking based speech separation, where the output context is explicitly employed to improve the accuracy of mask estimation. The key idea is to incorporate the estimated masks of several previous frames as additional inputs to better estimate the mask of the current frame. Rather than formulating it as a recurrent neural network (RNN), which is potentially much harder to train, we propose to train a deep neural network (DNN) with implicit deep stacking. The estimated masks of the previous frames are updated only at the end of each DNN training epoch, and then the updated estimated masks provide additional inputs to train the DNN in the next epoch. At the test stage, the DNN makes predictions sequentially in a recurrent fashion. In addition, we propose to use the L_1 loss for training. Experiments on the CHiME-2 (task-2) dataset demonstrate the effectiveness of our proposed approach.

Index Terms— deep stacking networks, recurrent neural networks, deep neural networks, speech separation

1. INTRODUCTION

For speech separation/enhancement in the short-time Fourier transform domain, the ideal solution is to obtain the clean magnitude and clean phase, with which clean signals can be re-synthesized perfectly. However, phase information is difficult to be estimated from noisy utterances. Therefore, many studies focus on recovering the clean magnitude and use the noisy phase for re-synthesis. Recently, deep learning has shown considerable potential for supervised speech separation [1], [2]. DNNs have been used to estimate an ideal time-frequency (T-F) mask [1], or directly map to clean features from noisy ones [2], [3]. In [4], Wang *et al*. carefully compare T-F masking and spectral mapping, and suggest that masking should be preferred.

In this study, we investigate how to leverage output patterns or output context information for better mask estimation. We emphasize that improving mask estimation can benefit a lot of tasks, such as speech de-noising [1], room de-reverberation [3], [5], [6], multi-talker speech separation [7], phase reconstruction [8], acoustic beamforming [9], [10], [11], [12], and robust automatic speech recognition [13], [14] and speaker recognition.

There are clearly strong output patterns in ideal binary or ratio masks. We believe that these output patterns can be potentially utilized to improve mask estimation, as the output patterns represent some kind of regularization that the estimated masks should conform with. In recent years, various neural networks have been employed for mask estimation, such as DNNs [1], convolutional neural nets (CNNs), recurrent neural nets (RNNs) [15] with long-short term memory (LSTM) [16], [17], but none of them explicitly utilizes output context for mask estimation.

The key idea of the proposed approach is to use the estimated masks of previous frames as additional inputs to predict the mask at the current frame. This is akin to incorporating an n-gram language model defined on the output patterns into traditional frame-level mask estimation. In this way, the contextual information in the output is explicitly utilized, and can be potentially modeled using an RNN. However, formulating it as an RNN would make the optimization process difficult, as the network would be very deep if we unfold the network through time for optimization. In addition, the output activation function in supervised separation is normally sigmoidal, likely leading to vanishing gradient problems during optimization.

We thus propose the recurrent deep stacking approach, in which the estimated masks of previous frames are updated at the end of every training epoch, and the updated estimated masks are then used as additional inputs to train the DNN in the next epoch. At the test stage, we need the estimated masks of several previous frames to predict the mask at the current frame. To obtain them, we formulate the DNN as an RNN to make predictions sequentially. The recurrent connections are from the output units of previous frames to the input of the current frame. In addition, we propose to use the L_1 loss for mask estimation. In terms of the SDR evaluation metric, the performance of our system is better than a strong LSTM result reported in [16] on the CHiME-2 (task-2) dataset.

The rest of this paper is organized as follows. We present our proposed method in Section 2. Experimental setup and results are presented in Sections 3 and 4. We conclude this paper in Section 5.

2. SYSTEM DESCRIPTION

In this section, we first describe the use of the L_1 loss for mask estimation and then present recurrent deep stacking networks for output context utilization.

2.1. Mask Estimation

The key idea of supervised speech separation is to use a supervised learning machine, such as DNNs, CNNs and LSTMs, to estimate the ideal ratio mask (IRM) [18] from a noisy utterance. With the estimated IRM, the clean magnitude can be reconstructed by point-wise multiplication in the time-frequency domain.

Traditionally, the square root of the Wiener filter is used as the IRM for training. In this study, we use a slightly different ideal mask as the training target [19], i.e.

978-1-5090-4117-6/17/$31.00 ©2017 IEEE

$$M_{t,f} = min\left(1, \frac{S_{t,f}^2}{Y_{t,f}^2}\right) \tag{1}$$

where $S_{t,f}^2$ and $Y_{t,f}^2$ represents the speech energy and mixture energy within a specific time-frequency bin, respectively. The values in this ideal mask are capped to be between 0 and 1, so that the mask values in different channels are bounded in the same range suitable for training, and sigmoid units can be utilized as the activation function at the output layer. The motivation for using this target is that after multiplying this ideal mask with the mixture power spectrogram, the resulting power spectrogram would be closer to the clean power spectrogram than using the Wiener filter or its square root variant.

Conventionally, mean square error, i.e. L_2 loss, is used for mask estimation. In this study, we propose to use L_1 loss as the loss function. Mathematically, the loss and its error gradient are defined in the following equations.

$$Loss = \frac{1}{T}\sum_t \sum_f \left|M_{t,f}^* - M_{t,f}\right| \tag{2}$$

$$\frac{\partial Loss}{\partial M_{t,f}^*} = \frac{1}{T}\left(1\left[M_{t,f}^* > M_{t,f}\right] - 1\left[M_{t,f}^* \leq M_{t,f}\right]\right) \tag{3}$$

where T is the total number of frames in the training data, $M_{t,f}^*$ represents the estimated mask at a specific T-F unit, and $1[\cdot]$ is the indicator function. By using the L_1 loss, we implicitly assume that the error term distribution is Laplacian [20]. We think that this assumption is reasonable considering the sparseness of speech and noise in the time-frequency domain, i.e., for many T-F units, only one source dominates. This is one of the reasons behind the ideal binary mask (IBM) notion [21]. Because of this property, the histogram of the ideal masks would be largely concentrated around 0 and 1, and exponentially decay from 0 and 1 to 0.5, at least when room reverberation is not considered. In such cases, it would be more reasonable to assume that the error term distribution would be Laplacian as well. In our experiments, we will demonstrate that if we use the L_1 loss for training, the error histogram on the validation set would be close to Laplacian, while if we use the L_2 loss for training, the error histogram on the validation set would not be similar to Gaussian.

After obtaining the estimated mask from a noisy utterance, we multiply it point-wisely with the noisy power spectrogram using Eq. (4) to get the enhanced power spectrogram.

$$\hat{Y}^2 = M^* \otimes Y^2 \tag{4}$$

where $\otimes$ represents point-wise matrix multiplication in the time-frequency domain. We use the noisy phase directly for re-synthesis.

2.2. Recurrent Deep Stacking Networks

Our model is essentially a DNN. The input is a combination of noisy features and the estimated masks of several previous frames, i.e.

$$< M_{t-w}^*, \ldots, M_{t-1}^*, x_{t-w}, \ldots, x_t, \ldots, x_{t+w} > \tag{5}$$

where w is the half-window length, and x_t and M_t^* represent the extracted noisy features and the estimated mask at frame t, respectively. The output is the ideal mask at the central frame, i.e. M_t. By using $M_{t-w}^*, \ldots, M_{t-1}^*$ as the additional inputs to predict M_t, the output context information is explicitly utilized for mask estimation. It is similar to including a $(w+1)$-gram language model defined on the output patterns into conventional frame-wise mask estimation approaches, with which we can have a strong belief on what M_t^* should be like after obtaining $M_{t-w}^*, \ldots, M_{t-1}^*$.

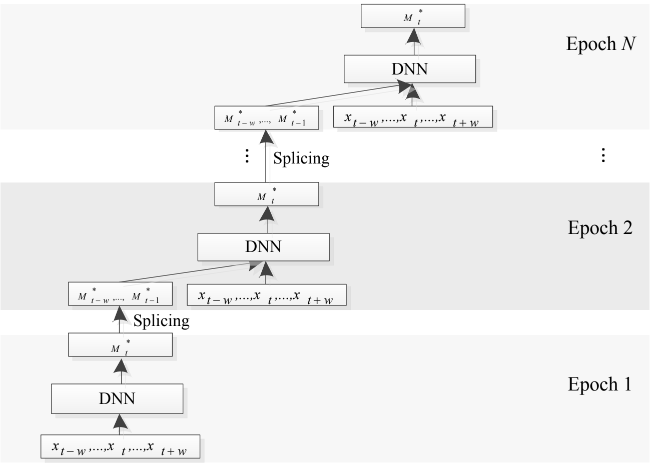
M_t^*
Epoch N
DNN
$M_{t-w}^*, \ldots, M_{t-1}^*$
$x_{t-w}, \ldots, x_t, \ldots, x_{t+w}$
Splicing
M_t^*
Epoch 2
DNN
$M_{t-w}^*, \ldots, M_{t-1}^*$
$x_{t-w}, \ldots, x_t, \ldots, x_{t+w}$
Splicing
M_t^*
Epoch 1
DNN
$x_{t-w}, \ldots, x_t, \ldots, x_{t+w}$

Fig. 1. Illustration of the training process of the proposed approach

The overall training process is shown in Fig. 1. We update all M_t^* at the end of every training epoch, and use the updated M_t^* as additional input features for DNN training in the following epoch. The effect is similar to implicitly stacking N DNNs, where N is the number of training epochs. The DNN model at each epoch is one module in the stack. In this way, a large context window at the input level can also be implicitly used, because the outputs of the DNN in the previous epoch are spliced together as additional inputs to the DNN in the current epoch, and each output is obtained by the previous DNN using multiple frames. Therefore, the more DNNs we stack, the more input context can be potentially utilized.

Although we stack a lot of DNNs at the training stage, there is no need to save all of them for testing. Interestingly, we only need to save the DNN model after the last training epoch. At the test stage, we formulate it as an RNN, where the recurrent connections are from the output units of the previous frames to the input of the current frame. This way, we can make predictions sequentially. More specifically, since our approach uses only past estimated masks, all the input features will be available when it comes to the current frame. When predicting the mask of the first frame, we just set all the estimated masks of the previous frames to zeros.

We point out that we can actually train the model as an RNN. However, it incurs many optimization difficulties, such as vanishing gradient problems as pointed out in the introduction section. In addition, training an RNN from the scratch is much slower than DNN training, because we have to move frame by frame in the forward and backward pass. Furthermore, the data shuffling in RNN training is not as good as that in DNN training due to its sequential nature. More importantly, more advanced DNN training techniques, such as batch normalization [22] and residual connections [23], can be easily incorporated into our method, while including these techniques into RNN or LSTM training may be quite difficult [24].

Several previous studies have applied deep stacking networks [25] to supervised speech separation [13], [26], [27], but only a limited number (two or three) of DNNs or several shallow networks are stacked. In these studies, each module in the stack is trained from the scratch using the outputs from lower modules together with original noisy features, and therefore each module has to go through a number of epochs for training. In our approach, we train our DNN for a fixed number of epochs, and the DNN model at each epoch is considered as an implicit module in the stack. Thus, a large number of modules can be stacked, and more context information in the input level can be utilized due to stacking. Most differently, all the stacked models in the previous studies have to be saved for testing, while only one model needs to be saved for our method. By formulating the trained DNN model as an RNN at the test stage, we can explicitly incorporate output context information into mask estimation.

The DNN in our study has four hidden layers, each with 2048 exponential linear units (ELUs) [28]. In our experiments, ELUs lead to faster convergence and better performance over the commonly used rectified linear units (ReLUs). The dropout rates of the input layer and all the hidden layers are set to 0.05. Besides the estimated masks of previous frames, we use log power spectrogram features with a symmetric 19-frame context window as the inputs, meaning w is set to 9. The window length is 25ms and the hop size is 10ms. We perform 512-point FFT when extracting log power spectrogram features. The dimension of the log power spectrogram features in our study is therefore 257, and so is the output dimension in our DNN. No pre-emphasis is performed before FFT. We find that using 25ms window length and 512-point FFT gives us consistently better results than using 20ms window length and 320-point FFT. All the features are globally mean-variance normalized before DNN training. We re-compute the mean and variance of the estimated masks after every update. Note that we need to feed all the training data to update the estimated masks after every training epoch. The network is trained using AdaGrad with a momentum term for 30 epochs. The learning rate is fixed at 0.005 in the first 10 epochs and linearly decreased to 10^{-4} in subsequent epochs. The momentum is linearly increased from 0.1 to 0.9 in the first 5 epochs and fixed at 0.9 afterwards.

3. EXPERIMENTAL SETUP

We conduct our experiments on the noisy and reverberant CHiME-2 dataset (task-2) [29] [1], rather than on our own manually mixed data as in many other studies. The major reason for choosing this dataset is that it is an open dataset, on which different studies and groups can fairly compare their baselines and results with each other. The reverberant and noisy signals are created by first convolving the clean signals in the WSJ0-5k corpus with binaural room impulse responses (BRIRs), and then adding reverberant noises recorded at six different SNR levels linearly spaced from -6 dB to 9 dB. The noises are recorded in a domestic living room and kitchen, which include a rich collection of sounds, such as electronic devices, background speakers, distant noises, footsteps, background music and so on. The BRIRs are recorded in the same environments. There are 7138 utterances in the training data (~14.5h in total), 409 utterances for each SNR level in the validation data (~4.5h in total), and 330 utterances for each SNR level in the test data (~4h in total).

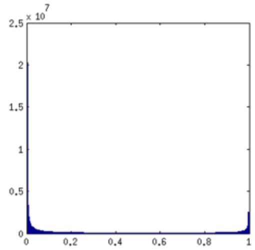

Fig. 2. The histogram of all the values in the ideal masks on the -6 dB subset of the validation set.

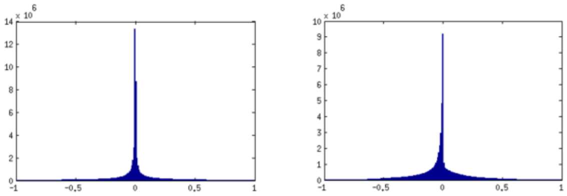

Fig. 3. Error histograms on the -6 dB subset of the validation set. The left histogram is obtained using the DNN trained with the L_1 loss, and the right histogram is obtained using the DNN trained with the L_2 loss.

Our system is monaural in nature. We merge the two-channel signals by a simple average. The effect is the same as applying delay-and-sum beamforming to the binaural signals, because the speaker is designed to be approximately in front of the two microphones. In our study, we use the averaged reverberant signals as the reference signals, so that we can construct ideal masks for DNN training, and calculate various evaluation metrics, such as the Short-Time Objective Intelligibility (STOI), Perceptual Estimation of Speech Quality (PESQ) and Signal-to-Distortion Ratio (SDR). The STOI and PESQ values are the objective measures of speech intelligibility and quality, respectively. Note that our model only tries to remove or attenuate additive noises.

4. EVALUATION RESULTS

We first compare the performance of the L_1 and L_2 loss for mask estimation, and then report the results of our proposed recurrent deep stacking networks. Finally, we compare our results with other studies in the literature.

4.1. Use of L_1 Loss for Mask Estimation

The comparison between the use of the L_1 and L_2 loss is presented in the second and third entries in Table I, II and III. We can clearly see that using the L_1 loss for DNN training leads to consistently better SDR, PESQ and STOI scores at all the six SNR levels. Note that in our experiments, we just change the loss functions for DNN training and fix all the other hyper-parameters in order to make a fair comparison. In Fig. 2, we plot the histogram of the ideal masks. Clearly, the distribution has two modes around 0 and 1, and exponentially decays towards the middle. In Fig. 3, we plot the histograms of the errors at all the time frequency units on the -6 dB subset of the validation set, one for each loss function. We can see that if we use the L_1 loss for training, the histogram is pretty similar to Laplacian distribution, which justifies the assumptions. In contrast, if we use the L_2 loss for training, the histogram is obviously not similar to Gaussian at all. We think that this explains why the L_1 loss leads to better performance in our experiments.

[1] Available at http://spandh.dcs.shef.ac.uk/chime_challenge/chime2013/WSJ0/ - as of August 2016.

TABLE I. COMPARISON OF SDR SCORES ON TEST SET

Approaches	Loss functions	-6dB	-3dB	0dB	3dB	6dB	9dB	Average
Unprocessed	-	-2.55	-1.12	1.11	2.78	4.48	5.78	1.75
DNN	L_2	8.94	10.42	12.28	13.90	15.60	17.51	13.11
DNN	L_1	9.76	11.12	12.88	14.43	16.05	17.89	13.69
Recurrent Deep Stacking Networks	L_1	10.35	11.70	13.43	14.91	16.46	18.25	14.18
Recurrent Deep Stacking Networks	+Signal Approximation	**10.76**	**12.06**	**13.69**	**15.08**	**16.57**	**18.33**	**14.41**
LSTM [16]	Signal Approximation	10.46	11.85	13.40	14.86	16.34	18.07	14.17

TABLE II. COMPARISON OF PESQ SCORES ON TEST SET

Approaches	Loss functions	-6dB	-3dB	0dB	3dB	6dB	9dB
Unprocessed	-	2.138	2.327	2.492	2.662	2.854	3.049
DNN	L_2	2.791	2.940	3.076	3.217	3.356	3.506
DNN	L_1	2.888	3.049	3.186	3.321	3.449	3.586
Recurrent Deep Stacking Networks	L_1	2.996	3.162	3.295	3.432	3.533	3.663
Recurrent Deep Stacking Networks	+Signal Approximation	**3.014**	**3.181**	**3.315**	**3.448**	**3.559**	**3.685**
Phoneme-specific Speech Separation [30]	Signal Approximation	2.731	2.884	3.011	3.146	3.284	3.430

TABLE III. COMPARISON OF STOI SCORES ON TEST SET

Approaches	Loss functions	-6dB	-3dB	0dB	3dB	6dB	9dB
Unprocessed	-	0.737	0.778	0.813	0.852	0.881	0.909
DNN	L_2	0.871	0.895	0.914	0.932	0.944	0.957
DNN	L_1	0.878	0.901	0.919	0.936	0.946	0.959
Recurrent Deep Stacking Networks	L_1	**0.886**	**0.909**	**0.925**	**0.940**	**0.950**	**0.961**
Recurrent Deep Stacking Networks	+Signal Approximation	0.884	0.907	0.924	0.939	0.948	0.959
Phoneme-specific Speech Separation [30]	Signal Approximation	0.861	0.886	0.905	0.922	0.935	0.949

4.2. Recurrent Deep Stacking Networks

We first train our recurrent deep stacking networks using the L_1 loss until convergence. Then we switch to the signal approximation loss used in [30] and further train the model until convergence. Note that in our experiments, training the model using the signal approximation loss from the scratch gives much worse performance than using the L_1 or the L_2 loss, as is suggested in the original paper [31]. By comparing the third and fourth entries in Table I, II and III, we can see that modeling output context explicitly leads to clear improvements especially in terms of SDR and PESQ scores. Further training the model using the signal approximation loss leads to better SDR and PESQ results while slightly worse STOI numbers.

We compare our methods with several other studies with experiments on the same dataset in the literature. All of them use log power spectrogram features. In [30], a phoneme-specific speech separation approach that utilizes the information from robust ASR systems is proposed. Their models for speech separation are a bunch of DNNs, one for each phoneme, trained with the signal approximation loss. Only STOI and PESQ scores are reported in their study. From the last two entries in Table II and III, we can see that our results are clearly better. The results reported in [16] represent a series of efforts [32], [31], [33], [34] by a combination of groups on the CHiME-2 dataset. Only SDR scores are reported to measure the performance of speech separation in their studies. As reported in the last two entries of Table I, our model obtains slightly better results than the strong LSTM model trained with the signal approximation loss reported in [16]. Here, it should be noted that in [16], better SDR results are reported by using phase information and information from a robust ASR system. As our major goal here is to estimate masks more accurately, we compare our results with the reported LSTM model trained with the signal approximation loss, which does not utilize extra sources of information.

5. CONCLUDING REMARKS

We have proposed recurrent deep stacking networks to explicitly incorporate contextual information in the output patterns for mask estimation. In addition, we have proposed to use the L_1 loss for mask estimation, which gives us consistently better results than the widely used L_2 loss. Experimental results on the CHiME-2 dataset (task-2) are encouraging. The proposed recurrent deep stacking algorithm can be incorporated into conventional RNNs and LSTMs as a way to leverage output information by replacing the DNNs in this study with RNNs or LSTMs. In this way, the error gradients can be directly propagated multiple frames backwards. It can also be applied to spectral mapping based speech enhancement, as clean spectrogram contains even stronger output patterns. From a wider viewpoint, it can be potentially applied to improve many other tasks, in which the output context provides useful constraints, such as acoustic modeling in automatic speech recognition and sequence labeling in natural language processing. One potential drawback of the proposed approach is that the input dimension is dependent on the output dimension. Nonetheless, the findings in this study suggest that, at a minimum, explicitly modeling output patterns would likely bring consistent improvements for time-frequency masking based supervised speech separation.

6. ACKNOWLEDGEMENTS

This research was supported in part by an AFRL contract (FA8750-15-1-0279), an NSF grant (IIS-1409431), and the Ohio Supercomputer Center.

7. REFERENCES

[1] Y. Wang and D.L. Wang, “Towards Scaling Up Classification-based Speech Separation,” *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 21, no. 7, pp. 1381–1390, 2013.

[2] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, “A Regression Approach to Speech Enhancement Based on Deep Neural Networks,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2015.

[3] K. Han, Y. Wang, D. L. Wang, W. S. Woods, and I. Merks, “Learning Spectral Mapping for Speech Dereverberation and Denoising,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 6, pp. 982–992, 2015.

[4] Y. Wang, A. Narayanan, and D. L. Wang, “On Training Targets for Supervised Speech Separation,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[5] D. Bagchi, M. Mandel, Z. Wang, Y. He, A. Plummer, and E. Fosler-Lussier, “Combining Spectral Feature Mapping and Multi-channel Model-based Source Separation for Noise-robust Automatic Speech Recognition,” in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2015.

[6] Y. Zhao, Z.-Q. Wang, and D. Wang, “A Two-stage Algorithm for Noisy and Reverberant Speech Enhancement,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, to appear.

[7] D. Yu, M. Kolbæk, Z. Tan, and J. Jensen, “Permutation Invariant Training of Deep Models for Speaker-independent Multi-talker Speech Separation,” in *arXiv preprint arXiv:1607.00325*, 2016.

[8] D. Williamson, Y. Wang, and D. L. Wang, “Complex Ratio Masking for Monaural Speech Separation,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, pp. 483–492, 2016.

[9] T. Yoshioka, N. Ito, M. Delcroix, A. Ogawa, K. Kinoshita, M. Fujimoto, C. Yu, W. J. Fabian, M. Espi, T. Higuchi, S. Araki, and T. Nakatani, “The NTT CHiME-3 System: Advances in Speech Enhancement and Recognition for Mobile Multi-microphone Devices,” in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2015, pp. 436–443.

[10] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, “Robust MVDR Beamforming using Time-frequency Masks for Online/Offline ASR in Noise,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 5210–5214.

[11] H. Erdogan, J. Hershey, S. Watanabe, and M. Mandel, “Improved MVDR Beamforming using Single-channel Mask Prediction Networks,” in *Proceedings of Interspeech*, 2016.

[12] X. Zhang, Z.-Q. Wang, and D. Wang, “A Speech Enhancement Algorithm by Iterating Single- and Multi-microphone Processing and its Application to Robust ASR,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, to appear.

[13] A. Narayanan and D. L. Wang, “Investigation of Speech Separation as a Front-end for Noise Robust Speech Recognition,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 4, pp. 826–835, Apr. 2014.

[14] Z.-Q. Wang and D. L. Wang, “A Joint Training Framework for Robust Automatic Speech Recognition,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 796–806, Apr. 2016.

[15] P. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, “Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, pp. 2136–2147, 2015.

[16] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller, “Speech Enhancement with LSTM Recurrent Neural Networks and its Application to Noise-robust ASR,” in *International Conference on Latent Variable Analysis and Signal Separation*, 2015, pp. 91–99.

[17] J. Chen and D. Wang, “Long Short-Term Memory for Speaker Generalization in Supervised Speech Separation,” in *Proceedings of Interspeech*, 2016.

[18] A. Narayanan and D. L. Wang, “Ideal Ratio Mask Estimation using Deep Neural Networks for Robust Speech Recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7092–7096.

[19] Y. Wang, A. Misra, and K. Chin, “Time-frequency Masking for Large Scale Robust Speech Recognition,” in *Proceedings of Interspeech*, 2015, pp. 2469–2473.

[20] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[21] D. L. Wang and G. J. Brown, *Eds., Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Hoboken, NJ: Wiley-IEEE Press, 2006.

[22] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” in *arXiv preprint arXiv:1502.03167*, 2015.

[23] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *arXiv preprint arXiv:1512.03385*, 2015.

[24] G. Pereyra, Y. Zhang, and Y. Bengio, “Batch Normalized Recurrent Neural Networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 2657–2661.

[25] B. Hutchinson, L. Deng, and D. Yu, “Tensor Deep Stacking Networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1944–1957, Aug. 2013.

[26] X.-L. Zhang and D. L. Wang, “A Deep Ensemble Learning Method for Monaural Speech Separation,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, pp. 967–977, 2016.

[27] X. Zhang, H. Zhang, S. Nie, and G. Gao, “A Pairwise Algorithm Using the Deep Stacking Network for Speech Separation and Pitch Estimation,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, pp. 1066–1078, 2016.

[28] D. Clevert, T. Unterthiner, and S. Hochreiter, “Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs),” in *arXiv preprint arXiv:1511.07289*, 2015.

[29] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, “The Second ‘CHiME’ Speech Separation and Recognition Challenge: An Overview of Challenge Systems and Outcomes,” in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 162–167.

[30] Z.-Q. Wang, Y. Zhao, and D. L. Wang, “Phoneme-specific Speech Separation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 146–150.

[31] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, “Discriminatively Trained Recurrent Neural Networks for Single-channel Speech Separation,” in *IEEE Global Conference on Signal and Information Processing*, 2014, pp. 577–581.

[32] F. Weninger, J. Le Roux, J. Hershey, and S. Watanabe, “Discriminative NMF and its Application to Single-channel Source Separation,” in *INTERSPEECH*, 2014, pp. 865–869.

[33] H. Erdogan, J. Hershey, S. Watanabe, and J. Le Roux, “Phase-sensitive and Recognition-boosted Speech Separation Using Deep Recurrent Neural Networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 708–712.

[34] Z. Chen, S. Watanabe, H. Erdogan, and J. Hershey, “Speech Enhancement and Recognition using Multi-task Learning of Long Short-term Memory Recurrent Neural Networks,” in *Proceedings of Interspeech*, 2015.