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

Speaker-aware neural network based beamformer for speaker extraction in speech mixtures

Kateřina Žmolíková[1,2], *Marc Delcroix*[1], *Keisuke Kinoshita*[1], *Takuya Higuchi*[1], *Atsunori Ogawa*[1], *Tomohiro Nakatani*[1]

[1]NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan
[2]Brno University of Technology, Speech@FIT, Czech Republic
izmolikova@fit.vutbr.cz, marc.delcroix@lab.ntt.co.jp

Abstract

In this work, we address the problem of extracting one target speaker from a multichannel mixture of speech. We use a neural network to estimate masks to extract the target speaker and derive beamformer filters using these masks, in a similar way as the recently proposed approach for extraction of speech in presence of noise. To overcome the permutation ambiguity of neural network mask estimation, which arises in presence of multiple speakers, we propose to inform the neural network about the target speaker so that it learns to follow the speaker characteristics through the utterance. We investigate and compare different methods of passing the speaker information to the network such as making one layer of the network dependent on speaker characteristics. Experiments on mixture of two speakers demonstrate that the proposed scheme can track and extract a target speaker for both closed and open speaker set cases.

Index Terms: speaker extraction, speaker-aware neural network, beamforming, mask estimation

1. Introduction

Extracting a speech signal of a target speaker from recordings corrupted by noise, reverberation and interfering speakers is an important and difficult problem. Although methods for suppressing background noise have recently advanced greatly, recovery of speech in presence of interfering speakers still remains a challenge. However, in many applications, such as meeting recognition, an overlap between multiple simultaneous speakers is frequent. Traditionally, research on overlapping speech enhancement aims at separating all speech signals active in the recordings. This problem has been investigated using both single and multi-channel source separation techniques, including NMF [1], ICA [2] or clustering of spatial cues [3, 4, 5].

In contrast to speech separation, speaker extraction aims at extracting a single target speaker from the background or speech mixture. Recently, there have been great advances in extracting speech from an observation in a noisy environment using a beamformer [6, 7, 8]. In particular, a scheme combining neural network based mask estimation together with beamforming was suggested by different works [7, 8] and appears particularly effective [9, 10]. In these approaches, a neural network estimates a time-frequency mask that extracts speech from the mixture of speech and noise. This mask is then employed to compute spatial covariance matrices (SCM) of speech and noise that are then used to derive the beamformer filters. Because of the great difference between time-frequency characteristics of speech and noise, the neural network is able to accurately estimate masks to extract speech.

Using the above-mentioned scheme in the case of multiple active speakers is, however, not straight-forward. Several works have addressed the mask estimation with neural networks in multi-speaker cases [11, 12, 13, 14, 15]. These works aim at recovering all active speakers from the mixture. This brings the problem of permutation ambiguity, e.g. if the network outputs masks for each speaker in the mixture, it is ambiguous which output of the network is associated to which speaker, and such association can change arbitrarily from a processing segment to another. Some works avoided this problem by extracting sources which are clearly distinguishable by gender [15] or dominance [14]. Such conditions may however not be fulfilled in real scenarios. The problem can be also alleviated by using long segments, though this disables the possibility of real-time processing and can hurt the performance of the methods [11, 12].

In this work, we explore an alternative approach where we inform the neural network about the target speaker, so that it extracts only his speech signal. This is accomplished by passing additional information to the network about the target speaker such as the speaker identity (in the case of closed-speaker set) or speaker characteristics extracted from a short *adaptation utterance* containing only the target speaker (in the case of open-speaker set). This scheme simply avoids the problem of permutation as the network learns to track the target speaker and outputs a mask for this speaker only. Furthermore, it also enables to track the speaker not only across different processing segments, but also more globally, e.g. different utterances or recording sessions. In addition, the architecture is independent of the number of speakers present in the mixture.

Different mechanisms for informing the network about the speaker for modifying its behavior were previously explored for the task of speaker adaptation. The most common approaches include using speaker representation as additional input feature [16, 17, 18] or adapting part of the parameters for each speaker [19, 20, 21]. A method leveraging advantages of both of these approaches was proposed in [22]. In this paper, we get inspired from speaker adaptation techniques developed for ASR to realize target speech extraction. Note that this work shares similarities with guided speech enhancement approaches which explore incorporating additional knowledge about the sources to improve the enhancement performance [23, 24].

The rest of the paper is structured as follows. In Section 2, we describe the scheme of neural network mask estimation for beamforming and its application to the multispeaker case. Section 3 describes our proposed method in detail. In Section 4, we describe the performed experiments and comment the results.

2. Neural network based mask estimation for beamforming

In this section, we review the scheme proposed in [7] for speaker extraction in presence of noise and discuss the difficulty of applying this to speaker extraction in speech mixtures.

http://dx.doi.org/10.21437/Interspeech.2017-667

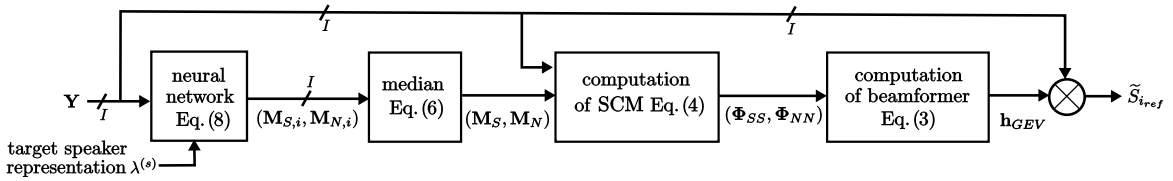

Figure 1: *The entire processing with multichannel signal $\mathbf{Y}_i$ as input and estimate of the target speaker signal $\widetilde{\mathbf{S}}_{i_{\mathrm{ref}}}$ as the output.*

2.1. Overview of the method

The observed microphone signal at the ith microphone in the STFT domain can be modeled as a summation of a desired source and a noise signal

$$Y_i(t,f) = S_i(t,f) + N_i(t,f), \tag{1}$$

where $i = 1,\dots,I$ is the index of microphone, t is the time frame, $f = 1,\dots,F$ is the frequency-bin index, $Y_i(t,f)$ is the observed signal, $S_i(t,f)$ is an image of the speech signal and $N_i(t,f)$ is the noise signal.

The beamforming process typically aims at estimating the image of the speech signal at reference microphone i_{ref}

$$\widetilde{S}_{i_{\mathrm{ref}}}(t,f) = \mathbf{h}^{\mathrm{H}}(f)\mathbf{Y}(t,f), \tag{2}$$

where $\widetilde{S}_{i_{\mathrm{ref}}}(t,f)$ is the estimated signal, $\mathbf{h}(f)$ is a vector of the beamformer coefficients and $\mathbf{Y}(t,f) = [Y_1(t,f)\dots Y_I(t,f)]$. Following [7], we use a Generalized Eigenvector beamformer (GEV), whose filters are obtained as

$$\mathbf{h}_{\mathrm{GEV}}(f) = \underset{\mathbf{h}(f)}{\arg\max}\, \frac{\mathbf{h}^{\mathrm{H}}(f)\mathbf{\Phi}_{SS}(f)\mathbf{h}(f)}{\mathbf{h}^{\mathrm{H}}(f)\mathbf{\Phi}_{NN}(f)\mathbf{h}(f)}, \tag{3}$$

where $\mathbf{\Phi}_{SS}(f)$ and $\mathbf{\Phi}_{NN}(f)$ are the spatial covariance matrices of speech and noise respectively. They can be obtained as

$$\mathbf{\Phi}_{vv}(f) = \sum_{t=1}^{T} M_v(t,f)\mathbf{Y}(t,f)\mathbf{Y}^{\mathrm{H}}(t,f), \tag{4}$$

where $v \in \{S,N\}$ and $M_v(t,f)$ denotes a time-frequency mask for speech or interference. In this work, we also employed a single-channel postfilter after beamforming as defined in [7].

The time-frequency masks $M_v(t)$ are estimated using a neural network. In particular, the neural network computes a mask for each channel and these masks are then combined using a median operation to get an overall mask as

$$(\mathbf{M}_{S,i}(t), \mathbf{M}_{N,i}(t)) = g(\mathbf{y}_i(t)), \tag{5}$$

$$\mathbf{M}_v(t) = \underset{i}{\mathrm{median}}(\mathbf{M}_{v,i}(t)), \tag{6}$$

where g is a transformation computed by the neural network, $\mathbf{y}_i(t) = [|Y_i(t,1)|\dots|Y_i(t,F)|]$ and $\mathbf{M}_{v,i}(t) = [M_{v,i}(t,1)\dots M_{v,i}(t,F)]$. The neural network is trained using ideal binary masks as targets, which are computed from the noisy and corresponding clean data.

2.2. Problem of neural network based speech-speech extraction

In contrast to the previous work, we aim to use the above scheme for extracting a target speaker from mixtures of multiple speakers. The observation model thus becomes

$$Y_i(t,f) = \sum_{j=0}^{J-1} S_i^{(j)}(t,f) + N_i(t,f), \tag{7}$$

where j is the index of the speaker. In our case, the mask $\mathbf{M}_S$ should thus cover the T-F points dominated by the target speaker, while $\mathbf{M}_N$ should cover the T-F points dominated by all interfering speakers or noise.

While in the speech-noise separation, the desired speech can be clearly distinguished from the noise due to its unique characteristics, it is not true when the interfering signal is speech. This ambiguity can be avoided if the target signal has a special characteristic as being the most dominant one [14] or having a specific gender [15]. However, if the target cannot be directly identified, the neural network estimating the masks has no information about which of the speakers is the desired one. In this case, the network learns to estimate mask $\mathbf{M}_S$ covering the speech from all speakers, not performing speaker extraction.

3. Proposed scheme

There are different ways to tackle the problem described in 2.2. In this work, we propose to inform the network about the identity of the target speaker. The processing of the neural network as shown in Eq. (5) thus becomes

$$(\mathbf{M}_{S,i}(t), \mathbf{M}_{N,i}(t)) = g(\mathbf{y}_i(t), \lambda^{(\mathbf{s})}), \tag{8}$$

where $\lambda^{(\mathbf{s})}$ is a vector representing the target speaker s. The whole processing chain is depicted in Figure 1. Informing the network about the target speaker enables the network to learn to track characteristics of the target speaker and thus solves the permutation ambiguity.

We investigate two ways to represent speakers, i.e. one-hot vectors and speaker posteriors. The one-hot vectors represent the speaker ID and are suitable for a closed-speaker set condition, i.e. when the test speakers have been seen during the training. Speaker posteriors are obtained from a short *adaptation utterance* containing the speech of the target speaker only. To get the posteriors from the adaptation utterance, we use a neural network trained to predict training speakers. This representation can map unseen test speakers to the training speakers and can thus generalize to the open-speaker condition, i.e. when the test speaker has not been seen during training.

Modifying the behavior of a neural network for different speakers has been previously explored for speaker adaptation of an acoustic model used for speech recognition. Most of the speaker adaptation methods make some of the parameters of the network specific for each speaker. Here, we explore several of these approaches. This section describes different ways of conditioning the function g on the speaker representation $\lambda^{(s)}$.

3.1. Speaker specific network

The extreme case of adapting the network for individual speakers is making all the parameters in the network speaker specific. Training speaker dependent networks for speaker extraction was already successfully used in previous works [25, 26]. However, training a separate network for the target speaker requires having sizable amount of data from this speaker, which is not very practical and is difficult to extend to the open speaker condition. In this work, we consider this case to confirm the capability of the neural network to extract a target speaker.

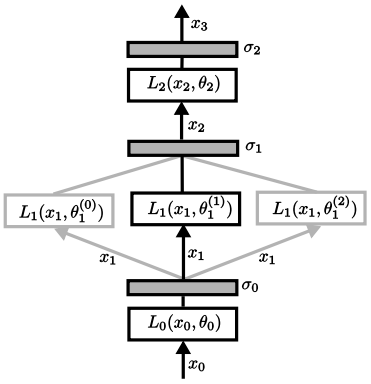

Figure 2: *Scheme of speaker specific layer configuration.*

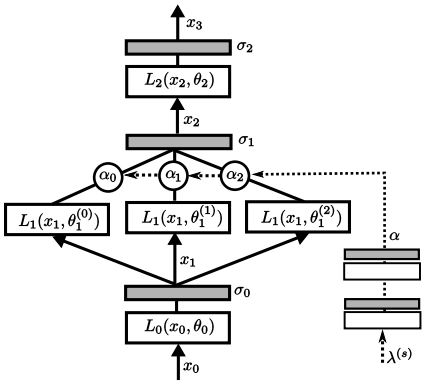

Figure 3: *Scheme of speaker adaptive layer configuration.*

3.2. Speaker specific bias

One of the methods shown to be efficient for speaker adaptation task is using the speaker representation as an additional feature either at the input or at one of the hidden layers of the network [16, 17, 18]. This effectively adapts the biases of this layer for each speaker. However, in our preliminary experiments we found this mechanism to be insufficient for the task of speaker extraction. The neural network with $\lambda^{(s)}$ as additional input of any of the layers converged to the same behavior as without any speaker information.

3.3. Speaker specific layer

Another speaker adaptation way which is more powerful than 3.2 but more practical than 3.1 is making one of the layers in the network speaker specific. This approach has been also explored for speaker adaptation in [19, 20]. This method is realized by dividing one of the layers of the network into several sub-layers, where each of these sub-layers correspond to one of the speakers in the training set. During both forward and backpropagation, only one of these layers is active at a time, i.e. the one corresponding to the speaker which is being processed.

Denoting the index of the speaker specific layer as k and index of the speaker being processed as s we can express the computation of the neural network as follows

$$\mathbf{x}_{n+1} = \begin{cases} \sigma_n(L_n(\mathbf{x}_n; \theta_n)) & \text{for} \quad n \neq k, \\ \sigma_n(L_n(\mathbf{x}_n; \theta_n^{(s)})) & \text{for} \quad n = k, \end{cases} \tag{9}$$

where $\mathbf{x}_n$ denotes the input to the nth layer, $L_n(\mathbf{x}, \theta)$ is the transformation computed by the nth layer parametrized by θ and σ_n is an activation function. For fully connected layers $\theta = \{\mathbf{W}, \mathbf{b}\}$ and $L(\mathbf{x}, \theta) = \mathbf{W}\mathbf{x} + \mathbf{b}$, where $\mathbf{W}$ is a weight matrix and $\mathbf{b}$ is a bias vector. Figure 2 shows an example of this configuration.

This approach is suitable for the case of a closed-speaker set, where the speaker identity is given. In this case, the speaker representation $\lambda^{(s)}$ can be a one-hot vector and the index s of the processed speaker can be directly inferred from $\lambda^{(s)}$. However, this approach is not directly extendable to the open-speaker condition.

3.4. Speaker adaptive layer

To overcome the above-mentioned drawback of a speaker-specific layer and be able to use a more versatile choice of the speaker representation $\lambda^{(s)}$, we employed a cluster adaptive training scheme, which has been also previously used for speaker adaptation [22].

With this approach, one of the layer is again divided into multiple sub-layers. Here, in contrast to the previous approach, the sub-layers do not correspond to the individual speakers. Instead, the output of the factorized layer is obtained as a weighted combination of the outputs of all the sub-layers. The weights used for this combination are specific for each of the speakers.

Following the previous notation we can express the computation of the neural network as

$$\mathbf{x}_{n+1} = \begin{cases} \sigma_n(L_n(\mathbf{x}_n; \theta_n)) & \text{for } n \neq k, \\ \sigma_n(\sum_{m=0}^{M-1} \alpha_m^{(s)} L_n(\mathbf{x}_n; \theta_n^{(m)})) & \text{for } n = k. \end{cases} \tag{10}$$

The speaker-specific weights $\alpha^{(s)}$ can be inferred using an auxiliary neural network which has a speaker representation $\lambda^{(s)}$ as its input. This auxiliary network can then be trained jointly with the main mask estimating network. The scheme of the configuration is depicted in Figure 3.

For unseen speakers, we can either extract $\lambda^{(s)}$ from the adaptation utterance and use it as input of the auxiliary network as in [22] or drop the auxiliary network and directly compute α with error backpropagation given an adaptation utterance as in [21]. In this case the adaptation utterance should be mixed with an interference and α is updated to optimize the ideal binary mask for this mixture.

4. Experiments

To compare the different methods we carried out experiments on simulated data. The accuracy was evaluated on the beamforming outputs with speech enhancement measures — cepstral distance (CD) [27] and signal-to-distortion ratio (SDR) [28]. In this section, we describe the created dataset, experimental settings and discuss the results.

4.1. Data

The simulated data is based on the Wall Street Journal dataset [29]. In particular, we used the training and test set partition used in the CHiME3 challenge [30] and mixed an interference utterance from different speaker into each of the utterances. This means that the training set consists of 7138 utterances from 83 speakers and the test set has 410 utterances from 10 speakers.

To create multichannel mixtures, we used room impulse responses generated with the image method [31, 32] with a circular microphone array with 8 microphones, 20 cm diameter and RT60=0.2 s. Two speakers are located at 1 or 1.5 m distance from the array in randomly selected angles from 0 to 180°.

In the experiments, we worked with two settings, i.e. a closed-speaker set and an open-speaker set. For the closed-speaker set, the 7138 training utterances were splitted into 90 % closed-train and 10 % closed-test such that all speakers in the closed-test are covered in closed-train. The interfering utterances were randomly selected such that no utterance in the closed-test occurs as target or interference in the closed-train.

Table 1: *Results of the closed-set experiments, showing improvements over the unprocessed signal quality (that is 5.04 dB in CD, 0.63 dB in SDR). Both for CD and SDR improvements, higher is better. SS ≡ speaker specific, SA ≡ speaker adaptive.*

model	same gender ΔCD /ΔSDR	diff gender ΔCD /ΔSDR	all ΔCD /ΔSDR
SS network	1.73 / 6.33	1.97 / 7.50	1.85 / 6.91
SS layer	1.82 / 6.75	2.08 / 7.77	1.96 / 7.25
SA layer, 1hot	1.70 / 6.37	2.04 / 7.95	1.87 / 7.15
SA layer, post	1.71 / 6.57	2.04 / 7.98	1.88 / 7.27

4.2. Settings

We used the same configuration for the mask-estimation neural network as in [7], which consists of one BLSTM layer, two fully connected layers with ReLU activations and one fully connected layer with sigmoid activations. The numbers of units in the four layers are 512-1024-1024-512, respectively. The input of the network is one frame of the magnitude spectrum coefficients of a single channel and the output is the corresponding target speech and an interference mask. The network was trained to optimize cross entropy with ideal binary masks. For the training, we used Adam optimization scheme [33].

4.3. Closed-speaker-set experiments

Table 1 summarizes the results from the experiments with the closed-speaker set. First, we carried out experiments with a speaker-specific network. The results show an improvement in both CD and SDR measures, which confirms the ability of the network to track the target speaker through the utterance.

In the following experiments, we investigated the speaker-specific layer setup. In this experiment, the second layer in the network was made speaker-specific. We can see an improvement compared to the speaker-specific network case. This suggests that sharing most of the parameters of the network among all the speakers and thus training them on more data than in speaker-specific network case, is beneficial.

Finally, we changed the architecture to the speaker adaptive layer to make the setup better applicable to different speaker representations. We again used the second layer in the network but factorized it into 30 sub-layers. The weights α were computed by an auxiliary network from a one-hot vector. Both the main and auxiliary networks were randomly initialized and jointly trained. The obtained improvement is very similar to the case of a speaker-specific layer. This means that we do not need to create a unique layer for each of the training speakers and opens the possibility of including unseen speakers by estimating the weights from a different speaker representation.

To investigate this option, we replaced the one-hot vectors with speaker posteriors obtained by separate neural network as described in Section 3 and retrained the main and auxiliary networks using the posteriors. The accuracy in this case was comparable with using one-hot vectors, which shows that the posteriors in the closed-speaker set case are sufficiently accurate.

4.4. Open-speaker-set experiments

Following the findings obtained with the closed set of speakers, we extended the experiments to speakers unseen in the training. The results from the experiments can be found in Table 2. Note that the results in Tables 1 and 2 cannot be directly compared as the closed-test set and open-test set differ.

In the first experiment, we derived the weights α using the auxiliary network with speaker posteriors derived from the adaptation utterance as an input. In this case, the network still managed to extract the target speaker in different gender mixtures, but on same gender mixtures, it mostly failed. This may be caused by an inaccurate estimation of α by the auxiliary network that was not trained on the unseen speaker.

Table 2: *Results of the open-set experiments, showing improvements over the unprocessed signal quality (that is 5.23 dB in CD, 0.17 dB in SDR). Both for CD and SDR improvements, the higher the better. SS ≡ speaker specific, SA ≡ speaker adaptive.*

model	same gender ΔCD /ΔSDR	diff gender ΔCD /ΔSDR	all ΔCD /ΔSDR
SA layer, post	0.52 / 1.46	2.01 / 7.56	1.28 / 4.57
SA layer, adapt α	1.19 / 4.13	2.05 / 7.33	1.63 / 5.76

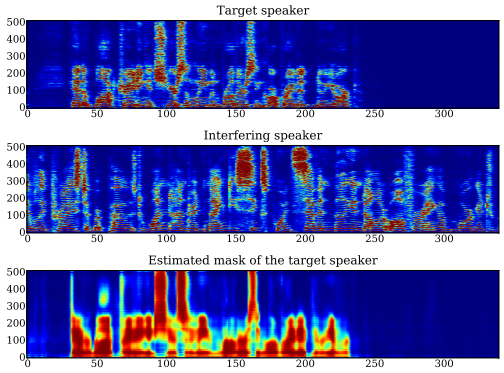

Figure 4: *Example of mask estimated in the open-speaker set.*

To confirm this, we tested a different way to estimate the weight α using the same adaptation data. That is instead of using the posterior extracted from the adaption-utterance on the input of the auxiliary network, we directly retrained the weights α using a mixture created by mixing the adaptation utterance with other speaker from the training set. In this case, the results improved also for same gender mixtures.

This result shows that even for unseen speakers, the speaker adaptive layer approach can extract the target speaker with a good estimation of α obtained using the direct adaptation approach. It is also noteworthy that the direct adaptation does not require any additional information for the adaptation although it may be less practical as it involves retraining for the unseen speaker. Figure 4 shows an example of an estimated mask in this case. This suggests that using a better speaker representation than posteriors on the input of auxiliary network and thus better estimating α could lead to a better extraction. This will be subject of our future investigations.

5. Conclusion

In this paper, we proposed a method for informing a neural network about a target speaker so that it can extract this speaker from a speech mixture. This enables to use a previously proposed neural network based beamformer scheme for multi-speaker case. The experiments show that making one layer in the network depend on a speaker representation allows tracking of the speaker. Although the investigations are still preliminary, this approach shows promising results. In future work, we plan to explore different speaker representations, optimizing the architecture or joint training with ASR.

6. Acknowledgment

Katerina Zmolikova was partly supported by Czech Ministry of Interior project No. VI20152020025 ”DRAPAK”, and by Czech Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project ”IT4Innovations excellence in science - LQ1602”.

7. References

[1] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 1–12, 2007.

[2] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent component analysis*. John Wiley & Sons, 2004, vol. 46.

[3] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on signal processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[4] H. Sawada, S. Araki, and S. Makino, "Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 516–527, 2011.

[5] M. I. Mandel, R. J. Weiss, and D. P. Ellis, "Model-based expectation-maximization source separation and localization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 382–394, 2010.

[6] T. Higuchi, N. Ito, S. Araki, T. Yoshioka, M. Delcroix, and T. Nakatani, "Online MVDR beamformer based on complex gaussian mixture model with spatial prior for noise robust ASR," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 4, pp. 780–793, 2017.

[7] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 2016, pp. 196–200.

[8] H. Erdogan, J. R. Hershey, S. Watanabe, M. Mandel, and J. Le Roux, "Improved MVDR beamforming using single-channel mask prediction networks," in *Interspeech*, 2016.

[9] J. Heymann, L. Drude, and R. Haeb-Umbach, "Wide residual BLSTM network with discriminative speaker adaptation for robust speech recognition."

[10] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech and Language*, 2016.

[11] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 2016, pp. 31–35.

[12] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 2017.

[13] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 2017.

[14] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep neural networks for single-channel multi-talker speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1670–1679, 2015.

[15] Y. Wang, J. Du, L. R. Dai, and C. H. Lee, "Unsupervised single-channel speech separation via deep neural network for different gender mixtures," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016, pp. 1–4.

[16] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 55–59.

[17] A. Senior and I. Lopez-Moreno, "Improving DNN speaker independence with i-vector inputs," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 225–229.

[18] V. Gupta, P. Kenny, P. Ouellet, and T. Stafylakis, "I-vector-based speaker adaptation of deep neural networks for french broadcast audio transcription," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 6334–6338.

[19] T. Ochiai, S. Matsuda, X. Lu, C. Hori, and S. Katagiri, "Speaker adaptive training using deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 6349–6353.

[20] R. Gemello, F. Mana, S. Scanzio, P. Laface, and R. De Mori, "Linear hidden transformations for adaptation of hybrid ANN/HMM models," *Speech Communication*, vol. 49, no. 10, pp. 827–835, 2007.

[21] C. Wu and M. J. F. Gales, "Multi-basis adaptive neural network for rapid adaptation in speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, 2015, pp. 4315–4319.

[22] M. Delcroix, K. Kinoshita, T. Hori, and T. Nakatani, "Context adaptive deep neural networks for fast acoustic model adaptation," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, 2015, pp. 4535–4539.

[23] E. Vincent, N. Bertin, R. Gribonval, and F. Bimbot, "From blind to guided audio source separation: How models and side information can improve the separation of sound," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 107–115, 2014.

[24] K. Kinoshita, M. Delcroix, A. Ogawa, and T. Nakatani, "Text-informed speech enhancement with deep neural networks," in *Interspeech*, 2015, pp. 1760–1764.

[25] J. Du, Y. Tu, Y. Xu, L. Dai, and C. H. Lee, "Speech separation of a target speaker based on deep neural networks," in *2014 12th International Conference on Signal Processing (ICSP)*, Oct 2014, pp. 473–477.

[26] X. L. Zhang and D. Wang, "A deep ensemble learning method for monaural speech separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 967–977, 2016.

[27] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2008.

[28] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[29] J. Garofolo, "CSR-I (WSJ0) Complete LDC93S6A," https://catalog.ldc.upenn.edu/ldc93s6a, 1993.

[30] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third CHiME speech separation and recognition challenge: Dataset, task and baselines," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 504–511.

[31] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[32] E. A. P. Habets, "Room impulse response generator," Technische Universiteit Eindhoven, Tech. Rep., 2010. [Online]. Available: http://home.tiscali.nl/ehabets/rir_generator/rir_generator.pdf

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980