SOURCE-AWARE CONTEXT NETWORK FOR SINGLE-CHANNEL MULTI-SPEAKER SPEECH SEPARATION

Zeng-Xi Li[⋆] *Yan Song*[⋆] *Li-Rong Dai*[⋆] *Ian McLoughlin*[†]

[⋆] National Engineering Laboratory for Speech and Language Information Processing, University of Science and Technology of China, Hefei, China.
[†] School of Computing, University of Kent, Medway, UK.

ABSTRACT

Deep learning based approaches have achieved promising performance in speaker-dependent single-channel multi-speaker speech separation. However, partly due to the label permutation problem, they may encounter difficulties in speaker-independent conditions. Recent methods address this problem by some assignment operations. Different from them, we propose a novel source-aware context network, which explicitly inputs speech sources as well as mixture signal. By exploiting the temporal dependency and continuity of the same source signal, the permutation order of outputs can be easily determined without any additional post-processing. Furthermore, a Multi-time-step Prediction Training strategy is proposed to address the mismatch between training and inference stages. Experimental results on benchmark WSJ0-2mix dataset revealed that our network outperformed state-of-the-art methods in both closed-set and open-set conditions with similar or comparable experimental settings, in terms of Signal-to-Distortion Ratio (SDR) improvement.

Index Terms— Speech Separation, Deep Learning, Label Permutation Problem

1. INTRODUCTION

Multi-speaker single-channel speech separation is the task of estimating the individual sources from a monaural mixture of speech. It has important real-world applications including robust automatic speech recognition, multi-speaker meeting transcription and audio/video captioning. Unlike human listeners who can concentrate on separate sources in an acoustic mixture, automatic speech separation is still a challenging and unsolved problem [1].

Over the past few decades, significant efforts have been devoted to speech separation [2, 3, 4, 5, 6, 7]. Before the emergence of deep learning, the most popular separation techniques were based on Computational Auditory Scene Analysis (CASA) [2, 3]. In CASA, segmentation and grouping rules, which are typically hand-engineered or heuristic, are utilized to group Time-Frequency (T-F) units belonging to the same speaker. Another technique is Non-negative Matrix Factorization (NMF) [4]. In NMF, a mixture is decomposed into specific activations using a set of non-negative bases. However, limited success has been achieved for multi-speaker separation. Recently, some deep learning approaches have been proposed, casting speech separation as a multi-class regression problem [5, 6, 7]. Despite their effectiveness at speaker-dependent separation, they often encounter difficulties with speaker-independent separation, partially in the *label permutation problem* [8], which will be detailed in Section 2.

More recently, several approaches have been proposed to address the label permutation problem, including the following: In [9, 10], instantaneous energy pattern or speaker information is used to determine speaker assignment. In [11, 8], Permutation Invariant Training (PIT) and utterance-level PIT determine the speaker order according to the lowest separation error within all possible permutations. In [12, 13], the Deep Clustering (DPCL) method performs label assignment using the clustering algorithm in a deep embedding space. In [14, 15], a Deep Attractor Network (DANet) creates attractor points in embedding space to determine the source assignment. The main idea of these methods is to determine source assignment based on similarity measurements in embedding space, or in the original spectral space (*e.g.*, distance of embedding vectors in DPCL and DANet, Mean Square Error (MSE) between estimated and target magnitude spectra in PIT).

Unlike those methods, we propose a novel source-aware context network for single-channel multi-speaker speech separation. As shown in Fig. 1, the proposed network explicitly inputs speech sources as well as mixture, and directly outputs estimated sources, which will be fed back as input sources for processing during the next time step. By exploiting dependency and continuity of the same source (important acoustic cues indicated by Auditory Scene Analysis (ASA) [2]), the permutation order of outputs can be easily determined without any additional operation. More details will be described in Section 3. A Multi-time-step Prediction Training (MPT)

This work was supported in part by National key research and development program (Grant No. 2017YFB1002200), and by National Natural Science Foundation of China (Grant No. U1613211).

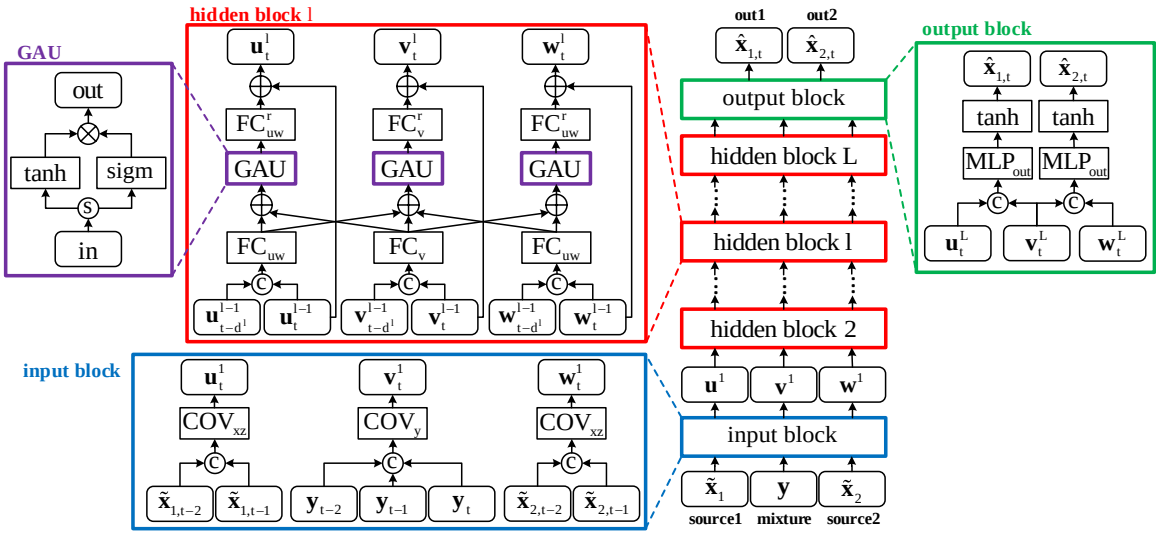

Fig. 1. An overview of the network G implementing Eqn. (4) for a two-speaker separation task. GAU, FC, MLP, sigm, ⓒ, Ⓢ, $\otimes$ and $\oplus$ represent Gated Activation Unit [16], full connection, multi layer perceptron, sigmoid activation, concatenation, equally slicing operation, element-wise multiplication and addition respectively.

strategy is further proposed to alleviate the mismatch between training and inference stages. Specifically, at each time step, network outputs are fed back as inputs for the next time step. This repeats across multiple time steps, generating a sequence of estimated sources in a recursive manner.

To evaluate the effectiveness of the proposed network, we conducted extensive experiments on the benchmark WSJ0-2mix dataset [12]. Results revealed that the MPT strategy allows our proposed network to outperform state-of-the-art methods in both closed-set and open-set conditions with similar or comparable experimental settings [12, 13, 14, 15, 11, 8].

2. LABEL PERMUTATION PROBLEM

As mentioned, conventional deep learning based methods commonly cast multi-speaker separation as a multi-class regression problem. For ease of description, we will focus on two-speaker situations [11, 8]. Generally, the separation model H can be formulated as

$$\hat{\mathbf{x}}_{1,t}, \hat{\mathbf{x}}_{2,t} = H(\mathbf{y}_{t+F}, \ldots, \mathbf{y}_{t-P}) \tag{1}$$

where $\hat{\mathbf{x}}_{1,t}$ and $\hat{\mathbf{x}}_{2,t}$ are the t-th estimated magnitude spectra of two sources, $\mathbf{y}_t$ is the observed t-th magnitude spectra of the mixture, which is generated as a waveform $y_n = x_{1,n} + x_{2,n}$, and the receptive field length of future and past spectra are denoted as F and P respectively. The model H is used to estimate the t-th clean magnitude spectra of the corresponding sources, *i.e.*, $\mathbf{x}_{1,t}$ and $\mathbf{x}_{2,t}$.

During training, the error between targets $[\mathbf{x}_{1,t}, \mathbf{x}_{2,t}]$ and outputs $[\hat{\mathbf{x}}_{1,t}, \hat{\mathbf{x}}_{2,t}]$ needs to be computed for back-propagation. However, for conventional deep learning based approaches, using only input $\mathbf{y}$, it is unknown in advance whether the outputs order is $[\hat{\mathbf{x}}_{1,t}, \hat{\mathbf{x}}_{2,t}]$ or $[\hat{\mathbf{x}}_{2,t}, \hat{\mathbf{x}}_{1,t}]$. This problem is referred to as the *label permutation problem* [8]. DPCL, DANet and PIT methods can all be represented by Eqn. (1). The difference is that the outputs order (and label permutation) is determined by additional operations such as assignment according to the lowest separation error, or clustering in a deep embedding space.

3. SOURCE-AWARE CONTEXT NETWORK

Unlike existing methods, our proposed model simultaneously and recursively estimates two sources by modeling the conditional distribution of current sources' spectra, given past sources' spectra and mixture spectra, *i.e.*,

$$\hat{\mathbf{x}}_{1,t} \sim p(\mathbf{x}_{1,t}|\mathbf{x}_{1,t-1}, \ldots, \mathbf{x}_{1,t-P}; \mathbf{x}_{2,t-1}, \ldots, \mathbf{x}_{2,t-P}; \mathbf{y}_t, \ldots, \mathbf{y}_{t-P}) \tag{2}$$

$$\hat{\mathbf{x}}_{2,t} \sim p(\mathbf{x}_{2,t}|\mathbf{x}_{2,t-1}, \ldots, \mathbf{x}_{2,t-P}; \mathbf{x}_{1,t-1}, \ldots, \mathbf{x}_{1,t-P}; \mathbf{y}_t, \ldots, \mathbf{y}_{t-P}) \tag{3}$$

Therefore, according to Eqns. (2)(3), the regression model G can be defined as

$$\hat{\mathbf{x}}_{1,t}, \hat{\mathbf{x}}_{2,t} = G(\tilde{\mathbf{x}}_{1,t-1}, \ldots, \tilde{\mathbf{x}}_{1,t-P}; \tilde{\mathbf{x}}_{2,t-1}, \ldots, \tilde{\mathbf{x}}_{2,t-P}; \mathbf{y}_t, \ldots, \mathbf{y}_{t-P}) \tag{4}$$

where $\tilde{\mathbf{x}}_{k,t'} = \hat{\mathbf{x}}_{k,t'}$, $\forall k = \{1, 2\}, t' = t - P, \ldots, t - 1$ during inference. As a result, without additional operations, the outputs order is determined in advance – just the same as input sources. It is worth noting that G does not require future mixture spectra during inference, and Eqn. (4) can be easily extended to multiple speaker separation tasks.

3.1. Network Architecture

Inspired by WaveNet [17], the network G implementing Eqn. (4) is designed as shown in Fig. 1. G consists of three parts: an input block, several stacked hidden blocks and an output block, which will be described below.

Input Block. In order to capture ASA acoustic cues like proximity in frequency and time, harmonicity, onset and offset [2], as shown with a blue rectangle in Fig. 1 and detailed in Table 1, the input block extracts local temporal and spectral features of neighboring spectra with 2D convolutional layers followed by PReLUs [18]. Experiments use an 8 kHz sample rate for all waveforms, from which μ-law companded [19] magnitude spectra of dimension 129 were computed at 32 ms frame length with 10 ms shift, resulting in a feature map width of 129 in the input layers.

Table 1. Input block details of $\mathrm{COV_y}$ and $\mathrm{COV_{xz}}$. Feature map shapes in Input and Reshape layers are denoted in (channel, height, width) format. Convolution settings are denoted in kernel–stride–pad–channel format.

Layers	$\mathrm{COV_y}$	$\mathrm{COV_{xz}}$
Input	(1,3,129)	(1,2,129)
Conv1	(2,8)–(1,4)–(0,2)–64	(2,8)–(1,4)–(0,2)–64
Conv2	(2,8)–(1,4)–(0,2)–128	(1,8)–(1,4)–(0,2)–128
Conv3	(1,4)–(1,2)–(0,1)–64	(1,4)–(1,2)–(0,1)–64
Reshape	(64,1,4) $\rightarrow$ (256,1,1)	(64,1,4) $\rightarrow$ (256,1,1)

Hidden Blocks are shown with red rectangles in Fig. 1. Hidden blocks are designed with the main consideration that the enhancement of the mixture will benefit from the estimation of clean sources, and vice versa. Using a conditioning method similar to [17], $\mathbf{u}_t^l$, $\mathbf{v}_t^l$ and $\mathbf{w}_t^l$ can be considered as outputs given conditions $\mathbf{v}_*^{l-1}$, $\{\mathbf{u}_*^{l-1}, \mathbf{w}_*^{l-1}\}$ and $\mathbf{v}_*^{l-1}$ respectively, *i.e.*,

$$\mathbf{u}_t^l = f(\mathbf{u}_t^{l-1}, \mathbf{u}_{t-d^l}^{l-1} | \mathbf{v}_t^{l-1}, \mathbf{v}_{t-d^l}^{l-1}) \tag{5}$$

$$\mathbf{w}_t^l = f(\mathbf{w}_t^{l-1}, \mathbf{w}_{t-d^l}^{l-1} | \mathbf{v}_t^{l-1}, \mathbf{v}_{t-d^l}^{l-1}) \tag{6}$$

$$\mathbf{v}_t^l = g(\mathbf{v}_t^{l-1}, \mathbf{v}_{t-d^l}^{l-1} | \mathbf{u}_t^{l-1}, \mathbf{u}_{t-d^l}^{l-1}, \mathbf{w}_t^{l-1}, \mathbf{w}_{t-d^l}^{l-1}) \tag{7}$$

where d^l is the temporal dilation factor for hidden block l. As long as L and d^l of all hidden blocks are known, receptive field length can be determined by $P = \Sigma d^l + 2$.

Output Block. As shown with a green rectangle in Fig. 1, outputs $\hat{\mathbf{x}}_{1,t}$ and $\hat{\mathbf{x}}_{2,t}$ are derived from $\{\mathbf{u}_t^L, \mathbf{v}_t^L\}$ and $\{\mathbf{w}_t^L, \mathbf{v}_t^L\}$ of the last hidden block L respectively using a multi layer perceptron structure equipped with a PReLU and a $\tanh$ activation, which consists of a 512-unit hidden layer and a 129-dimension output layer.

It is worth mentioning that, with respect to sources $\mathbf{x}_1$ and $\mathbf{x}_2$, network parameters are all shared, and the structure is completely symmetric. This characteristic conforms to the common sense that all positions of each source are equivalent and exchangeable. Moreover, this design avoids model size growth when source number increases.

3.2. Multi-time-step Prediction Training

Using a conventional training method such as [17], a mismatch problem would arise between training and inference stages. During training, source inputs $\tilde{\mathbf{x}}_{k,<t}$ in Eqn.(4) are clean spectra $\mathbf{x}_{k,<t}$, whereas in the inference stage, they change to estimated spectra $\hat{\mathbf{x}}_{k,<t}$. The error between the two spectra leads to a mismatch, especially prevalent when both sources show similar patterns (*e.g.*, energy, onset time and pitch trajectory).

To alleviate this problem, we adopt a Multi-time-step Prediction Training (MPT) strategy. At the first time step $t' = t$, source inputs are all clean spectra. Then at each time step t', outputs $\hat{\mathbf{x}}_{k,t'}$ are fed back as inputs $\tilde{\mathbf{x}}_{k,t'}$ to replace original clean spectra $\mathbf{x}_{k,t'}$ for the next time step $t'+1$. This procedure repeats S times recursively, generating a sequence of estimated source spectra $\hat{\mathbf{x}}_{k,t'}$, $t' = t, \ldots, t+S-1$. Finally, network parameters are optimized to minimize the averaged MSE between targets and corresponding estimated source spectra across all time steps:

$$L = \frac{1}{FS} \sum_{s=0}^{S-1} \sum_{k=1}^{K} \|\mathbf{x}_{k,t+s} - \hat{\mathbf{x}}_{k,t+s}\|_2^2 \tag{8}$$

where S, K and F are step number, source number and frequency bin number respectively (in this paper K=2, F=129), $\|\cdot\|_2$ is the L_2 norm.

4. EXPERIMENTS

4.1. Experimental Setup

We evaluated our network on the WSJ0-2mix dataset, which was also used in [12, 14, 8]. WSJ0-2mix was introduced in [12] and is derived from the WSJ0 corpus [20]. A 30-hour training set and a 10-hour validation set contained two-speaker mixtures generated by utterances randomly selected from the WSJ0 training set `si_tr_s`, which were mixed at various Signal-to-Noise Ratios (SNR) between 0 dB and 10 dB. A 5-hour test set was similarly generated using utterances from 16 speakers in the WSJ0 development set `si_dt_05` and evaluation set `si_et_05`. The validation set and the test set were used to evaluate separation performance for closed condition (CC) and open condition (OC) (unseen speaker) respectively.

The network G evaluated had 10 hidden blocks with dilation factors $[d^2, \ldots, d^{11}]$=[1, 2, 4, 8, 16, 1, 2, 4, 8, 16] and dimensions of $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$ were set to 256. As a result, G had approximately 7.2 million parameters, and from Section 3.1, we can see that P=64.

To accelerate training with MPT described in Section 3.2, a strategy was employed following curriculum learning [21]: The training starts from small S in Eqn. (8), when validation loss converges, S increases to a larger value. In the experiments, this procedure repeats several times with S increasing in the order of 1, 5, 10, 30, 60, 90 and 120, where S=1 is equivalent to conventional training method. The network was optimized using the Adam algorithm [22] with learning rate 0.001 for S=1, 0.0002 for S=5, and 0.0001 for other S settings. Meanwhile, dropout (p=0.2) was only applied in the input block when $S < 60$.

Considering G is completely symmetric with respect to two sources $\mathbf{x}_1$ and $\mathbf{x}_2$, during inference, source inputs $\tilde{\mathbf{x}}_{1,<t}$ and $\tilde{\mathbf{x}}_{2,<t}$ cannot all be initialized to silent spectra, otherwise two outputs $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ will always be the same. Instead, in the experiments $\tilde{\mathbf{x}}_{1,<t}$ are initialized to $P-5=59$ frames of silent spectra plus 5 frames of mixture spectra $\mathbf{y}$, while $\tilde{\mathbf{x}}_{1,<t}$ are all initialized to P frames of silent spectra.

All experiments were implemented using MXNet [23], with the separated waveform reconstructed from the estimated sources' magnitude spectra, using phase from the original mixed speech.

4.2. Experimental Results

Separation performance was evaluated in terms of averaged Signal-to-Distortion Ratio (SDR) [24] improvement between separated speech and mixture.

Evaluation of Multi-time-step Prediction Training. We evaluate MPT with different step number S for closed condition (CC) and open condition (OC) respectively. The results are shown in Table 2.

We can see that conventional training methods (S=1) performed poorly, suggesting a large mismatch between training and inference. However, with MPT ($S > 1$), SDR was significantly improved, *e.g.*, 6.9 dB in OC when $S = 5$. Moreover, the best result (9.5 dB in OC) can be observed at S=60, which indicates that an appropriate ratio of clean and estimated source inputs may be essential for training. Finally, comparable or higher SDRs for OC compared to CC indicate that our network generalized well for unseen speakers.

Comparison with other methods. Table 3 summarizes SDR improvements and approximate model size (in terms of parameter number) for different methods with similar or comparable experimental settings. It is worth noting that all deep learning based models needed future mixture as input during inference, except for the last three models. We can see that the proposed network achieved comparable or higher SDR than DPCL, DANet and PIT models in OC and CC conditions with competitive model size.

In [8, 15], there are significant performance gaps between BLSTM and LSTM. For example, in [15], DANet-6 anchor achieved 10.4 dB SDR, higher than DANet-6 anchor-LSTM (9.0 dB SDR). This is reasonable since BLSTM can exploit future context information compared with LSTM. It is worth noting that our proposed network can be easily modified to incorporate future context as in BLSTM. In future we propose a detailed evaluation of parameter settings.

Table 2. Closed-condition (CC) and open-condition (OC) SDR improvements (dB) for different step numbers S in MPT. S=1 denotes conventional training method.

S	SDR Improvement	
	CC	OC
1	-3.0	-2.4
5	6.7	6.9
10	7.1	7.4
30	8.8	9.0
60	9.3	9.5
90	9.2	9.2
120	9.0	9.0

Table 3. SDR improvements (dB) and approximate model sizes (in terms of parameter number estimated according to the papers) of different methods for the same scenarios.

Method	Model Size (million)	SDR Imp. CC	SDR Imp. OC
Oracle NMF [12]	–	5.1	–
CASA [12]	–	2.9	3.1
DPCL [12]	6.3	6.5	6.5
DPCL+ [13]	10.6	–	9.4
PIT-CNN-51\51 [8]	–	7.6	7.5
uPIT-BLSTM-AM [8]	46.4	9.0	8.7
uPIT-BLSTM-PSM [8]	46.4	9.4	9.4
DANet-6 anchor-LSTM [15]	–	–	9.0
uPIT-LSTM-PSM [8]	65.7	7.0	7.0
ours	7.2	9.3	9.5

5. CONCLUSION

This paper proposed a novel source-aware context network and MPT strategy for single-channel multi-speaker speech separation. The network is designed to address the label permutation problem by exploiting temporal dependencies and continuity of the same speech source. During inference, no future mixture or post-processing is needed, making it more practical for on-line systems. The MPT strategy is further proposed to address the mismatch problem between training and inference stages. Experimental results on benchmark WSJ0-2mix revealed that, equipped with MPT strategy, our network outperformed state-of-the-art methods in both closed-set and open-set conditions with similar or comparable experimental settings, in terms of SDR improvements.

6. REFERENCES

[1] M. Cooke, J. R. Hershey, and S. J. Rennie, "Monaural speech separation and recognition challenge," *Computer Speech & Language*, vol. 24, no. 1, pp. 1–15, 2010.

[2] G. J. Brown and M. Cooke, "Computational auditory scene analysis," *Computer Speech & Language*, vol. 8, no. 4, pp. 297–336, 1994.

[3] Y. Shao and D. Wang, "Model-based sequential organization in cochannel speech," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 14, no. 1, pp. 289–298, 2006.

[4] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *INTERSPEECH*, 2006.

[5] J. Du, Y. Tu, Y. Xu, L. Dai, and C.-H. Lee, "Speech separation of a target speaker based on deep neural networks," in *Int. Conf. on Signal Process.* IEEE, 2014, pp. 473–477.

[6] Y. Tu, J. Du, Y. Xu, L. Dai, and C.-H. Lee, "Speech separation based on improved deep neural networks with dual outputs of speech features for both target and interfering speakers," in *Int. Symposium on Chinese Spoken Lang. Process.* IEEE, 2014, pp. 250–254.

[7] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE Trans. on Audio, Speech and Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[8] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 25, no. 10, pp. 1901–1913, 2017.

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

[10] Y. Wang, J. Du, L.-R. Dai, and C.-H. Lee, "A gender mixture detection approach to unsupervised single-channel speech separation based on deep neural networks," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 25, no. 7, pp. 1535–1546, 2017.

[11] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *IEEE Int. Conf. on Acoust., Speech and Signal Process.* IEEE, 2017, pp. 241–245.

[12] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *IEEE Int. Conf. on Acoust., Speech and Signal Process.* IEEE, 2016, pp. 31–35.

[13] Y. Isik, J. Le Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," *INTERSPEECH*, pp. 545–549, 2016.

[14] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *IEEE Int. Conf. on Acoust., Speech and Signal Process.* IEEE, 2017, pp. 246–250.

[15] ——, "Speaker-independent speech separation with deep attractor network," *arXiv preprint arXiv:1707.03634*, 2017.

[16] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves *et al.*, "Conditional image generation with pixelcnn decoders," in *Advances in Neural Inform. Process. Systems*, 2016, pp. 4790–4798.

[17] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *CoRR abs/1609.03499*, 2016.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. of the IEEE Int. Conf. on Comp. Vision*, 2015, pp. 1026–1034.

[19] C. Recommendation, "Pulse code modulation (pcm) of voice frequencies," *ITU*, 1988.

[20] J. Garofolo, D. Graff, D. Paul, and D. Pallett, "Csr-i (wsj0) complete ldc93s6a," *Philadelphia, USA: Linguistic Data Consortium*, 1993.

[21] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proc. of the 26th annual Int. Conf. on Machine Learning*. ACM, 2009, pp. 41–48.

[22] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang, "Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems," *arXiv preprint arXiv:1512.01274*, 2015.

[24] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. on Audio, speech, and Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.