INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Neural Speaker Extraction with Speaker-Speech Cross-Attention Network

Wupeng Wang, Chenglin Xu, Meng Ge, Haizhou Li

Department of Electrical and Computer Engineering, National University of Singapore, Singapore
e0125301@u.nus.edu, xuchenglin28@gmail.com, gemeng@tju.edu.cn, haizhou.li@nus.edu.sg

Abstract

In this paper, we propose a novel time-domain speaker-speech cross-attention network as a variant of SpEx [1] architecture, that features speaker-speech cross-attention. The speaker-speech cross-attention network consists of speech semantic layers that capture the high-level dependency of audio feature, and cross-attention layers that fuse speaker embedding and speech features to estimate the speaker mask. We implement cross-attention layers with both parallel and sequential concatenation techniques. Experiments show that the proposed models consistently outperform the state-of-the-art time-domain speaker extraction baseline on WSJ0-2mix dataset.

Index Terms: speaker extraction, speaker-speech cross-attention, multi-scale, time domain

1. Introduction

Cocktail party effect as formulated by Cherry [2] revealed that humans have the ability to attend to one target speaker while filtering out others in a multi-talker scenario. Neuro-psychologists believe that human brain uses an acoustic stimulus to reinforce the perceptual attractor, that is referred to as auditory selective attention [3]. There have been various attempts to emulate such human ability in automatic speech recognition [4–6], speaker verification [7], and speech diarization [8], through a speech separation or speaker extraction front-end.

Hershey et al. [9–11] proposed deep clustering (DC) technique to map mixture audio features into separable embedding space with deep neural networks. Chen et al. [12] implemented a deep attractor network (DANet) to generate specific masks to filter out corresponding outputs. To solve the permutation problem, Yu et al. [13] introduced permutation invariant training (PIT) strategy. Luo et al. [14–16] replaced the traditional short-time fourier transformation into learnable 1D convolution, that is referred to as time-domain audio separation network (TasNet). Chen et al. [17] further improved the model with transformer encoder structure. Despite much progress, the fact that the speech separation task seeks to separate all speech sources in the mixture makes it a challenging task in practice, for example, when a varying or unknown number of speakers are involved.

Speaker extraction is alternative to speech separation, which only extracts the voice of interest one at a time using a reference audio. The content of the reference audio is independent of the speech mixture, as long as it comes from the target speaker. There have been time-frequency domain solutions [18–20] which require magnitude and phase estimation. As phase estimation is not straightforward, Xu et al. [1] studied a speaker extraction network (SpEx) with a time-domain structure. Ge et al. [21] further extended the model with dynamic speaker embedding and twin encoder, that is known as SpEx+. The SpEx network architecture, with an effective masking mechanism, represents a typical neural speaker extraction approach.

Psychoacoustician Treisman's study [22] revealed that human auditory attention mechanism is able to identify a stimulus by, either physical properties or semantic information. In this paper, we propose a speaker-speech cross-attention network (CAN) that emulates such human ability, and make a major improvement over SpEx+ [21]. The main contributions of this paper include: 1) We propose the cross-attention layers to fuse the speaker information with speech content. 2) We propose the speech semantic layers to capture the long-range semantic content of the entire utterance instead of dilation mechanism in temporal convolutional network (TCN) [1].

The rest of the paper is organized as follows. Section 2 summarizes the problem in the SpEx+ approach. Section 3 introduces the proposed speaker-speech cross-attention network. In Section 4, we report the experiments. Finally, Section 5 concludes the paper.

2. Inactive Weight Problem in SpEx+

While [1] and [21] achieved impressive speaker extraction results in terms of signal-to-distortion ratio improvement (SDRi) and scale-invariant signal-to-distortion ratio improvement (SI-SDRi), they have not benefited from the utterance-level content information due to the limited receptive field of convolution kernels [17].

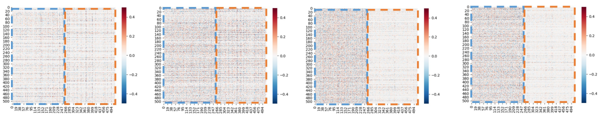
The speaker information is also not fully utilized in the deep TCN blocks. In [21], the neural speaker extraction network consists of four cascaded TCN blocks, where the sequence of frame-level speech features Z are concatenated with the speaker embedding matrix E to form the input to the TCN block. E represents a sequence of repeated speaker embedding vector e, and has the same length as Z. The first 1x1 convolutional neural network (1x1 CNN) of each TCN block performs the fusion between speaker embedding and speech features.

$$\begin{aligned} F_{\text{TCN}} &= \text{Conv1D}([E, Z]) \\ &= \begin{bmatrix} E, Z \end{bmatrix} \begin{bmatrix} W_E \\ W_Z \end{bmatrix} \\ &= EW_E + ZW_Z \end{aligned} \tag{1}$$

where Conv1D is the 1D convolution. W_E and W_Z are the fusion weights for speaker embedding and speech features.

We discover that Conv1D has not sufficiently fused the speaker-speech features due to the fact that the speaker embedding is only involved as a fixed fusion bias EW_E of the current frame, as opposed to a span of multiple frames. The speaker-speech fusion relies on the learnable weights W_Z and W_E, we plot the weights of the 1×1 convolution kernels in Fig. 1 for all four TCN blocks to visualize. It is noted that the weights for the 3^{rd} and 4^{th} TCN blocks in the dotted orange boxes, which attend to the speaker embedding, are inactive with near zero values as illustrated in Fig. 1(c) and Fig. 1(d). This suggests that they don't make use of speaker information.

http://dx.doi.org/10.21437/Interspeech.2021-2260



(a) 1^{st} TCN (b) 2^{nd} TCN (c) 3^{rd} TCN (d) 4^{th} TCN

Figure 1: *The weights of* 1×1 *convolutional kernels of 4 temporal convolutional network (TCN) blocks in SpEx+ [21] model. The kernel is used to fuse speech features and speaker embedding (see Eq. 1). The blue boxes denote the* W_Z *and the orange boxes denote the* W_E*. We observe that the* 1^{st} *and* 2^{nd} *TCN blocks attend to the speaker embedding inputs well, but the weights* W_E *to speaker embedding in the* 3^{rd} *and* 4^{th} *TCNs are inactive.*

3. Speaker Extraction with Speaker-Speech Cross-Attention Network

We propose a novel time-domain speaker extraction network, that features the speaker-speech cross-attention, to predict the mask for target speaker. The network includes four components: twin multi-scale speech encoder, speaker encoder, cross-attention speech extractor, and multi-scale speech decoder. For an effective comparison with SpEx+ network [21], we keep all components the same as in it except that we replace last two stacked TCN blocks with two stacked cross-attention network (CAN) blocks . The overall architecture is illustrated in Fig. 2.

3.1. Twin Multi-scale Speech Encoder

The mixture speech signal is encoded with learnable 1D convolution at multiple time scales to capture complementary information, e.g., phonemic and prosodic information over a short time span, and linguistic and semantic content over a long span across the utterance [23]. To encode the mixture and reference signals into same feature space, a twin multi-scale speech encoder is employed by sharing weights as [21] .

3.2. Speaker Encoder

Following [21], the same speaker encoder network is adopted to characterize the target speaker with a speaker embedding from the reference audio signal, as shown in Figure 2. The speaker encoder is optimized via multi-task learning with gradients from both the SI-SDR loss for speech extraction and the cross-entropy loss for speaker classification.

3.3. Cross-Attention Speech Extractor

The cross-attention speech extractor seeks to estimate the mask M_1,M_2 and M_3 at three different scales. The extractor takes in both the speech embedding matrix Y generated by the twin multi-scale speech encoder and the speaker embedding vector e derived from the speaker encoder. It consists of two stacked TCN blocks, and two stacked CAN blocks as illustrated in Fig. 2. The TCN structure follows what is in [21], as shown in Fig. 3(a). The CAN block is proposed to address the inactive weight problem for speaker embedding input in deep structure as discussed in Section 2. A stacked CAN block consists of stacked cross-attention layers and speech semantic layers as shown in Fig. 3(b). The former seeks to improve the interaction between speaker and speech features, while the latter seeks to model the utterance-level semantic content.

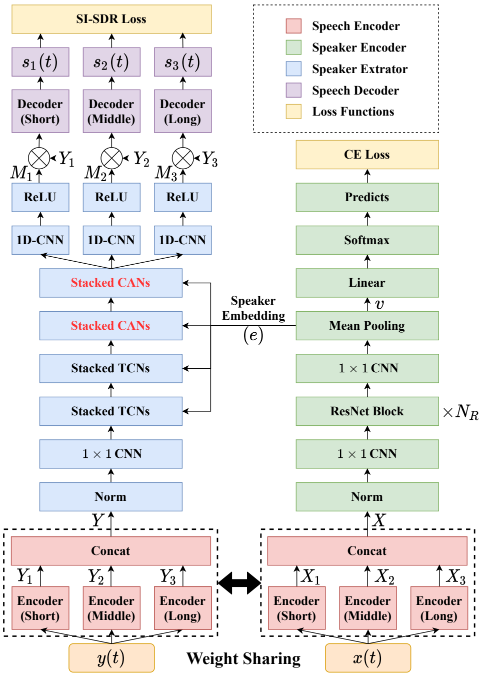


Figure 2: *The architecture of the speaker extraction network with speaker-speech cross-attention, where* $y(t)$ *and* $x(t)$ *are the input mixture and reference audio, respectively.* $Y = concat(Y_1, Y_2, Y_3)$ *denotes the speech features at three scales.* $X = concat(X_1, X_2, X_3)$ *denotes the reference features, while* M_1, M_2, M_3 *are the estimated masks.*

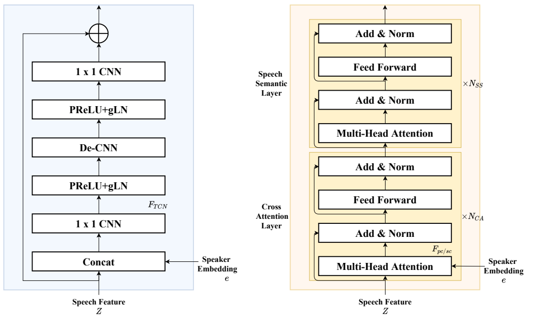
3.3.1. *Cross-Attention Layer with Parallel Concatenation*

We design a cross-attention network based on transformer [17] that takes speaker-speech embedding through *parallel concatenation* as in Fig. 4(a), that is denoted as pc. The speaker embedding vector e is repeatedly concatenated with speech features Z together to become the input to the cross-attention layer. We denote the repeated speaker embedding matrix as E. Then we can get the query Q_{pc}, the key K_{pc}, and the value V_{pc} to do multi-head attention as follows,

$$\begin{aligned}
Q_{\text{pc}} &= [E, Z]\begin{bmatrix} W_{\text{pc}}^{Q_E} \\ W_{\text{pc}}^{Q_Z} \end{bmatrix} = EW_{\text{pc}}^{Q_E} + ZW_{\text{pc}}^{Q_Z} \\
K_{\text{pc}} &= [E, Z]\begin{bmatrix} W_{\text{pc}}^{K_E} \\ W_{\text{pc}}^{K_Z} \end{bmatrix} = EW_{\text{pc}}^{K_E} + ZW_{\text{pc}}^{K_Z} \\
V_{\text{pc}} &= [E, Z]\begin{bmatrix} W_{\text{pc}}^{V_E} \\ W_{\text{pc}}^{V_Z} \end{bmatrix} = EW_{\text{pc}}^{V_E} + ZW_{\text{pc}}^{V_Z}
\end{aligned} \tag{2}$$

Here $W_{\text{pc}}^{Q_E}$,$W_{\text{pc}}^{K_E}$ and $W_{\text{pc}}^{V_E}$ are the transformation matrices of speaker embedding matrix E, respectively. $W_{\text{pc}}^{Q_Z}$,$W_{\text{pc}}^{K_Z}$ and $W_{\text{pc}}^{V_Z}$ are the transformation matrices of speech features Z. Next we can get speaker-speech cross-attention weight with scale factor d as following:

$$\begin{aligned}
A_{\text{pc}} &= softmax(Q_{\text{pc}}(K_{\text{pc}})^T/\sqrt{d}) \\
&= softmax((EW_{\text{pc}}^{Q_E}(W_{\text{pc}}^{K_E})^T E^T + EW_{\text{pc}}^{Q_E}(W_{\text{pc}}^{K_Z})^T Z^T \\
& ZW_{\text{pc}}^{Q_Z}(W_{\text{pc}}^{K_Z})^T Z^T + ZW_{\text{pc}}^{Q_Z}(W_{\text{pc}}^{K_E})^T E^T)/\sqrt{d})
\end{aligned} \tag{3}$$



(a) TCN (b) CAN

Figure 3: *A TCN block and a CAN block are illustrated in Fig. 3(a) and Fig. 3(b) respectively. Each CAN consists of the N_{CA} cross-attention layers and N_{SS} speech semantic layers. pc and sc represent two mechanisms to fuse the speaker embedding and mixture speech representations, named as the parallel concatenation and sequential concatenation, respectively.*

The attention weight A_{pc} is then applied to the value V_{pc} to obtain the attention head output F_{pc}.

$$\begin{aligned} F_{\text{pc}} &= A_{\text{pc}}V_{\text{pc}} \\ &= A_{\text{pc}}(EW_{\text{pc}}^{V_E} + ZW_{\text{pc}}^{V_Z}) \end{aligned} \tag{4}$$

Finally the attention head outputs F_{pc} of all heads are concatenated together and fed into the feed forword network. We also employ a residual connection around both multi-head attention and feed forword network, followed by layer normalization as in [17]. With the cross-attention mechanism, the attention weight A_{pc} is involved in the attention head output F_{pc} as in Eq. 4. When the attention weight A_{pc} is an identity matrix, the F_{pc} degenerates to F_{TCN}. The speaker embedding E contributes to both attention weight A_{pc} by the speaker-speech cross-attention $ZW_{\text{pc}}^{Q_Z}(W_{\text{pc}}^{K_E})^TE^T$ and $EW_{\text{pc}}^{Q_E}(W_{\text{pc}}^{K_Z})^TZ^T$ items and fusion bias $EW_{\text{pc}}^{V_E}$.

3.3.2. Cross-Attention Layer with Sequential Concatenation

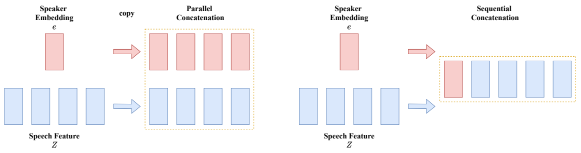
According to Eq. 3, we note what we need is the speaker-speech cross-attention terms. Therefore, instead of parallel concatenation, which doubles the parameters, we simply sequentially concatenates, denoted as *sequential concatenation*, the speaker embedding vector e to the speech feature matrix Z as shown in Fig. 4(b), to form an input the cross-attention layer. We denote this sequential concatenation cross-attention layer as sc, and has the attention weight A_{sc} as follows,

$$A_{\text{sc}} = softmax(\begin{bmatrix} eW_{\text{sc}}^Q(W_{\text{sc}}^K)^Te^T, eW_{\text{sc}}^Q(W_{\text{sc}}^K)^TZ^T \\ ZW_{\text{sc}}^Q(W_{\text{sc}}^K)^Te^T, ZW_{\text{sc}}^Q(W_{\text{sc}}^K)^TZ^T \end{bmatrix}/\sqrt{d}) \tag{5}$$

And the attention head output F_{sc} can be expressed as follows:

$$F_{\text{sc}} = A_{\text{sc}}\begin{bmatrix} eW_{\text{sc}}^V \\ ZW_{\text{sc}}^V \end{bmatrix} \tag{6}$$

Where W_{sc}^Q,W_{sc}^K and W_{sc}^V are the transformation matrices shared by speech feature Z and speaker embedding vector e. With the sequential concatenation, the number of learnable weights of sequential concatenation method is half of that of the parallel concatenation, but the speaker embedding continues to contribute to both attention weight A_{sc} and fusion bias eW_{sc}^V.



(a) Parallel Concatenation (b) Sequential Concatenation

Figure 4: *The left panel shows the traditional way of concatenation, that is to repeat the speaker embedding with every speech features, or parallel concatenation. The right panel shows the proposed sequential concatenation.*

3.3.3. Speech Semantic Layer

The transformer [24] has been shown effective in capturing long-range semantics in natural language processing [25, 26] and speech processing tasks [17, 27, 28]. Here we adopt the transformer architecture as the speech semantic layer to capture long-range semantics without speaker embedding and denote as ss. We follow the network architecture proposed by Chen et al. [17]. The full structure consists of a multi-head attention layer and feed forward network. Each attention head of multi-head attention layer first generates the query Q_{ss}, the key K_{ss}, and the value V_{ss} by a linear transformation.

$$Q_{\text{ss}} = ZW_{\text{ss}}^Q, K_{\text{ss}} = ZW_{\text{ss}}^K, V_{\text{ss}} = ZW_{\text{ss}}^V \tag{7}$$

where Z is the speech feature of current layer. $W_{\text{ss}}^Q, W_{\text{ss}}^K, W_{\text{ss}}^V$ are the transformation matrices. The attention weight A_{ss} depends on the dot-product attention with scaling factor d.

$$\begin{aligned} A_{\text{ss}} &= softmax(Q_{\text{ss}}K_{\text{ss}}^T/\sqrt{d}) \\ &= softmax(ZW_{\text{ss}}^Q(W_{\text{ss}}^K)^TZ^T/\sqrt{d}) \end{aligned} \tag{8}$$

Then the attention head output F_{ss} is the summation of value layer output V_{ss} multiplied by attention weight A_{ss}.

$$\begin{aligned} F_{\text{ss}} &= A_{\text{ss}}V_{\text{ss}} \\ &= A_{\text{ss}}ZW_{\text{ss}}^V \end{aligned} \tag{9}$$

Finally the attention head outputs F_{pc} of all heads are also concatenated and fed into the feed forward network. Unlike the fixed receptive field of convolution kernels in TCN block, which is unable to fully use the long-range semantics [16], our attention head output F_{ss} utilizes the utterance-level global semantic information through the full sequence attention in Eq. 9.

3.4. Multi-scale Speech Decoder

The cross-attention speech extractor predicts a mask for each of the three time scales. We apply each mask to the speech embedding Y via element-wise multiplication to extract the target speech and implement multi-scale speech decoder to reconstruct the time domain speech signal.

3.5. Learning Strategy

The weight multi-objective loss is adopted to optimize the speaker extraction network via multi-task learning.

$$\mathcal{L} = \mathcal{L}_{\text{SI-SDR}} + \lambda\mathcal{L}_{\text{CE}} \tag{10}$$

Table 1: *SDRi (dB) and SI-SDRi (dB) for different duration of reference speech on WSJ0-2mix.“7.3s (avg.)” refers to the average duration by randomly choosing one reference speech sample. “60s” refers to the fixed duration by combing several random chosen reference speech samples.*

Methods	Ref. Duration	SDRi (dB)	SI-SDRi (dB)
SpEx [1]	7.3s (avg.)	16.3	15.8
	60s	17.0	16.6
SpEx+ [21]	7.3s (avg.)	17.2	16.9
	60s	17.6	17.4
SpEx$_{\text{pc}}$	7.3s (avg.)	18.8	18.6
	60s	19.2	19.0
SpEx$_{\text{sc}}$	7.3s (avg.)	18.6	18.4
	60s	19.0	18.8

Table 2: *Speech separation and speaker extraction performance (dB) on the WSJ0-2mix dataset. For blind speech separation (BSS), we report the results evaluated on the oracle-selected outputs. For speaker extraction (SE), we report the average performance on each extracted stream of two speakers.*

Task	Methods	#Params	SDRi (dB)	SI-SDRi (dB)
BSS	cuPIT-Grid-RD [29]	53.2M	10.2	-
	SDC-G-MTL [30]	53.9M	10.5	-
	CBLDNN-GAT [31]	39.5M	11.0	-
	Chimera++ [11]	32.9M	12.0	11.5
	WA-MISI-5 [32]	32.9M	13.1	12.6
	BLSTM-TasNet [14]	23.6M	13.6	13.2
	Conv-TasNet [15]	5.1M	15.6	15.3
	DPRNN-TasNet [16]	2.6M	19.0	18.8
	DPT [17]	2.6M	20.6	20.2
	Wavesplit [33]	29M	22.2	22.3
SE	SpEx [1]	10.8M	17.0	16.6
	SpEx+ [21]	11.1M	17.6	17.4
	SpEx$_{\text{pc}}$	28.4M	19.2	19.0
	SpEx$_{\text{sc}}$	22.4M	19.0	18.8

where $\mathcal{L}_{\text{SI-SDR}}$ is the multi-scale scale-invariant signal-to-distortion ratio between the target clean speech and the reconstructed speech at three scales. Since the signals are reconstructed in three different scales, we also apply different weights to the SI-SDR calculated at each scale as [21]. $\mathcal{L}_{\text{CE}}$ is the cross-entropy loss for speaker classification. λ is the weight.

4. Experiments

We simulated WSJ0-2mix dataset[1] with 8kHz sampling rate based on WSJ0 corpus. The simulated database were divided into three sets: training set (20,000 utterances, 101 speakers), development set (5,000 utterances, 101 speakers), and test set (3,000 utterances, 18 speakers). For the training set and development set, the utterances from two speakers in WSJ0 “si_tr_s” corpus were randomly selected and mixed at a relative signal-to-noise ratio between 0 to 5 dB. Similarly, the utterances of test set were generated by WSJ0 “si_dt_05” and “si_et_05” subset. The 18 speakers in test set were not involved in the training set. The reference speech of the target speaker is randomly selected, that is different from the target utterance in the mixture.

4.1. Experimental Setup

We trained our models for 200 epochs on 2-second long segments. The learning rate was initialized as 1e-5 and decays by 0.5 when the loss on validation set was not improved in 2 consecutive epochs. Early stopping was applied if no better model was found in the validation set for 6 consecutive epochs. We utilized Adam optimizer for back-propagation of the full model. The size of twin multi-scale speech encoder and decoder kernels were 2.5ms, 10ms and 20ms, respectively. The number of ResNet blocks in speaker encoder N_R was 3, and the speaker embedding dimension was 256 in practice. We employed 4 parallel attention layers to do multi-head attention. The scale factor d was equal to the input feature dimension. The number of cross-attention layer N_{CA} and speech semantic layer N_{SS} in speaker extractor are 2, respectively. The weights for each scale in $\mathcal{L}_{\text{SI-SDR}}$ are kept same as in [21]. The weight λ in Eq. 10 is set to 10.

4.2. Experimental Results

We denote the network in Fig. 2 with parallel concatenation speaker aware layer as SpEx$_{\text{pc}}$, and the network with sequential concatenation speaker aware layer as SpEx$_{\text{sc}}$. We first compare the models with different reference duration as reported in Table 1. With average 7.3 seconds reference speech, the SpEx$_{\text{pc}}$ outperforms SpEx+ by 1.6 dB and 1.7dB in terms of SDRi and SI-SDRi, respectively. With 60 seconds reference speech, SpEx$_{\text{pc}}$ outperforms by 1.6 dB and 1.6dB as well. These results suggest that our new model achieves a consistent relative improvement over the state-of-the-art speaker extraction network.

SpEx$_{\text{pc}}$ slightly outperforms the SpEx$_{\text{sc}}$ with both 7.3 seconds and 60 seconds reference speech. The results suggest that the sharing of transformation matrix in Eq. 5 and Eq. 6 for sequential concatenation is not as effective as that for parallel concatenation.

We further compare several blind speech separation techniques and speaker extraction models on WSJ0-2mix dataset and report in Table 2. The proposed models are on par with DPRNN [16] but still behind DPT [17] and wavesplit [33]. We consider the performance gap is due to the difference in the extractor network, where we don't use dual path processing and dynamic mixing. We believe that the proposed ideas can be implemented with dual path processing and dynamic mixing, that will be our future work.

5. Conclusion

In this paper, we proposed a speaker extraction network with speaker-speech cross-attention structure. We further investigated how the speaker information could be well exploited in deep structure to better facilitate the speaker extraction. Experiments show that our proposed model achieves significant performance improvement over other state-of-the-art models.

6. Acknowledgements

This research work is supported by Programmatic Grant No. A1687b0033 from the Singapore Government's Research, Innovation and Enterprise 2020 plan (Advanced Manufacturing and Engineering domain). Project Title: Neuromorphic Computing.

[1] https://www.merl.com/demos/deep-clustering

7. References

[1] C. Xu, W. Rao, E. S. Chng, and H. Li, "Spex: Multi-scale time domain speaker extraction network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1370–1384, 2020.

[2] E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *The Journal of the acoustical society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[3] D. E. Broadbent, *Perception and communication*. Elsevier, 2013.

[4] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, "Robust automatic speech recognition: a bridge to practical applications," 2015.

[5] S. Watanabe, M. Delcroix, F. Metze, and J. R. Hershey, *New Era for Robust Speech Recognition*. Springer, 2017.

[6] X. Xiao, C. Xu, Z. Zhang, S. Zhao, S. Sun, S. Watanabe, L. Wang, L. Xie, D. L. Jones, E. S. Chng *et al.*, "A study of learning based beamforming methods for speech recognition," in *CHiME 2016 workshop*, 2016, pp. 26–31.

[7] W. Rao, C. Xu, E. S. Chng, and H. Li, "Target speaker extraction for overlapped multi-talker speaker verification," *arXiv preprint arXiv:1902.02546*, 2019.

[8] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe *et al.*, "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge." in *Interspeech*, 2018, pp. 2808–2812.

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

[10] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," *arXiv preprint arXiv:1607.02173*, 2016.

[11] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Alternative objective functions for deep clustering," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 686–690.

[12] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 246–250.

[13] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 241–245.

[14] Y. Luo and N. Mesgarani, "Real-time single-channel dereverberation and separation with time-domain audio separation network." in *Interspeech*, 2018, pp. 342–346.

[15] ——, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[16] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 46–50.

[17] J. Chen, Q. Mao, and D. Liu, "Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation," *arXiv preprint arXiv:2007.13975*, 2020.

[18] M. Delcroix, K. Zmolikova, K. Kinoshita, A. Ogawa, and T. Nakatani, "Single channel target speaker extraction and recognition with speaker beam," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5554–5558.

[19] C. Xu, W. Rao, E. S. Chng, and H. Li, "Optimization of speaker extraction neural network with magnitude and temporal spectrum approximation loss," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6990–6994.

[20] J. Wang, J. Chen, D. Su, L. Chen, M. Yu, Y. Qian, and D. Yu, "Deep extractor network for target speaker recovery from single channel speech mixtures," *arXiv preprint arXiv:1807.08974*, 2018.

[21] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "Spex+: A complete time domain speaker extraction network," *arXiv preprint arXiv:2005.04686*, 2020.

[22] A. M. Treisman, "Selective attention in man," *British medical bulletin*, vol. 20, no. 1, pp. 12–16, 1964.

[23] D. T. Toledano, M. P. Fernández-Gallego, and A. Lozano-Diez, "Multi-resolution speech analysis for automatic speech recognition using deep neural networks: Experiments on timit," *PloS one*, vol. 13, no. 10, p. e0205355, 2018.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[26] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[27] J. Kim, M. El-Khamy, and J. Lee, "T-gsa: Transformer with gaussian-weighted self-attention for speech enhancement," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6649–6653.

[28] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7829–7833.

[29] C. Xu, W. Rao, X. Xiao, E. S. Chng, and H. Li, "Single channel speech separation with constrained utterance level permutation invariant training using grid lstm," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6–10.

[30] C. Xu, W. Rao, E. S. Chng, and H. Li, "A shifted delta coefficient objective for monaural speech separation using multi-task learning." in *INTERSPEECH*, 2018, pp. 3479–3483.

[31] C. Li, L. Zhu, S. Xu, P. Gao, and B. Xu, "Cbldnn-based speaker-independent speech separation via generative adversarial training," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 711–715.

[32] Z.-Q. Wang, J. L. Roux, D. Wang, and J. R. Hershey, "End-to-end speech separation with unfolded iterative phase reconstruction," *arXiv preprint arXiv:1804.10204*, 2018.

[33] N. Zeghidour and D. Grangier, "Wavesplit: End-to-end speech separation by speaker clustering," *arXiv preprint arXiv:2002.08933*, 2020.