INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

SEF-Net: Speaker Embedding Free Target Speaker Extraction Network

Bang Zeng[1,2], *Hongbin Suo*[3], *Yulong Wan*[3], *Ming Li*[1,2†]

[1]School of Computer Science, Wuhan University, Wuhan, China
[2]Data Science Research Center, Duke Kunshan University, Kunshan, China
[3]Data&AI Engineering System, OPPO, Beijing, China

bangzeng@whu.edu.cn, ming.li369@dukekunshan.edu.cn

Abstract

Most target speaker extraction methods use the target speaker embedding as reference information. However, the speaker embedding extracted by a speaker recognition module may not be optimal for the target speaker extraction tasks. In this paper, we proposes Speaker Embedding Free target speaker extraction Network (SEF-Net), a novel target speaker extraction model without relying on speaker embedding. SEF-Net uses cross multi-head attention in the transformer decoder to implicitly utilize the speaker information in the reference speech's conformer encoding outputs. Experimental results show that our proposed model achieves comparable performance to other target speaker extraction models. SEF-Net provides a feasible new solution to perform target speaker extraction without using a speaker embedding extractor or speaker recognition loss function.

Index Terms: Target speaker extraction, speaker embedding free, dual-path, conformer

1. Introduction

Speech separation, also known as cocktail party problem [1], aims to recover all speaker's components from the mixed speech. Traditional speech separation methods [2, 3] assume the components statistically independent that may not hold in real-world. With the rapid development of Deep Neural Networks (DNN), many DNN-based methods are proposed, such as Deep Clustering (DPCL) [4, 5], Deep Attractor Network (DANet) [6], and Permutation Invariant Training (PIT) [7, 8]. However, these methods have an upper bound on reconstructing waves because of the utilization of Short-Time Fourier Transform [9] (STFT). To address this problem, Audio Source Separation (Tas-Net) [9, 10] and Dual-Path RNN (DPRNN) [11] directly process the mixed wave in time-domain. Recently, the transformer-based methods, such as dual-path transformer [12] network (DPT-Net) [13] and Sepformer [14], have become popular. Among them, Sepformer adopts a dual-path architecture like in [11]. Sepformer first splits the input sequence into smaller blocks and independently processes intra- and inter-block information. This approach is highly effective, resulting in state-of-the-art performance for speech separation [14]. However, despite their excellent performance, the unknown number of speakers poses practical challenges for applying speech separation models.

To solve this problem, target speaker extraction methods utilize a reference audio to extract the target speaker's component from the mixed wave. As shown in Figure 1(A), a typical time-domain target speaker extraction model relys on the target speaker embedding from a pre-trained [15, 16, 17] or joint-learned [18, 19] embedding extractor. However, the aim of training this extractor is usually to maximize speaker recognition performance. It may results that the speaker embedding is not optimal for the target speaker extraction task. Recently, there have been many studies to address this embedding mismatch issue. [20] investigates the specific role of speaker embedding in extracting target speakers. [21] compares various metric learning methods and emphasize the importance of distinctive speaker embedding. However, these methods still require to extract speaker embeddings and concatenate with input features frame-by-frame.

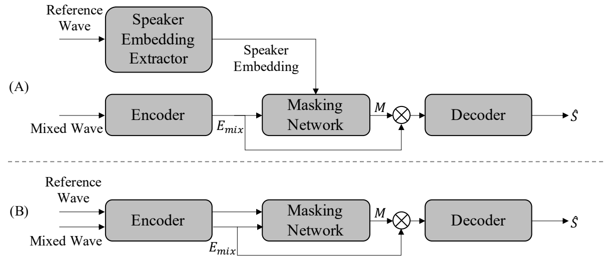

Figure 1: *(A) is the diagram of a typical time-domain target speaker extraction method. (B) is the diagram of our proposed method. ⊗ is an operation for element-wise product.*

In this paper, we propose Speaker Embedding Free target speaker extraction Network (SEF-Net), a novel target speaker extraction model. The abstract diagram of SEF-Net is shown in Figure 1(B). Unlike previous speaker extraction methods, SEF-Net does not need the target speaker embedding from a pre-trained or additional joint-trained speaker embedding extractor. In the masking network of SEF-Net, we first use two twin conformer [22] encoders to process the mixed and reference wave separatly. We apply weights sharing strategy [19] on these two twin conformer encoders. We thought that the conformer encodeing of the reference wave contains enough speaker information to extract the target speaker. Then the refernece wave's conformer encoding is used to query the mixed wave's conformer encoding in a tranformer decoder. In this case, the cross multi-head attention layer of transformer decoder acts as a feature fusion module. Our proposed method provides a new approach to handle the speaker embedding mismatch problem in target speaker extraction tasks. To the best of our knowledge, this work is the first to extract the target speaker's speech in time-domain without utilizing speaker embedding or related speaker recognition loss function.

The rest of this work is organized as follows. In Section 2, we present the SEF-Net's architecture. In Section 3, we report the experimental setup. In Section 4, we report the results and discussions. The conclusions are drawn in Section 5.

† Corresponding Author, ming.li369@dukekunshan.edu.cn

10.21437/Interspeech.2023-1749

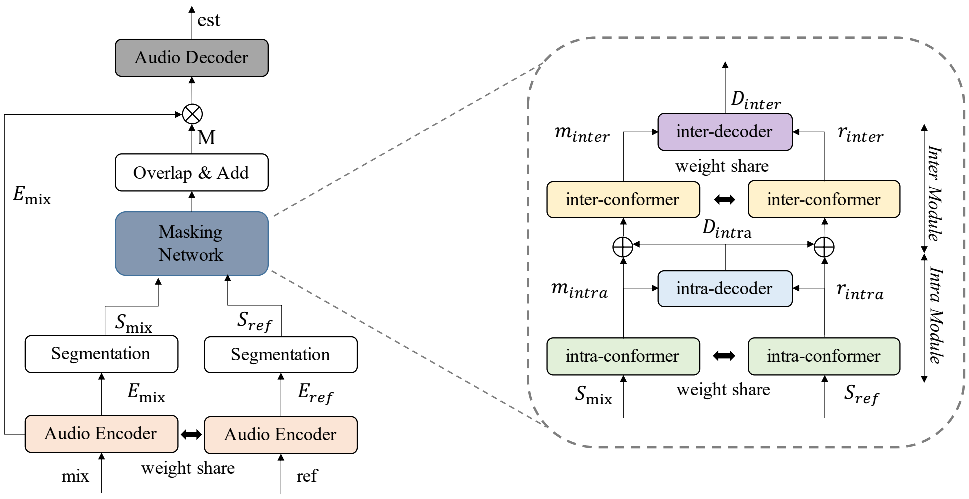

Figure 2: *The diagram of SEF-Net. We apply the weights sharing strategy on two twin audio encoders, intra-conformer, and inter-conformer blocks. mix and ref denote the mixed and reference wave, respectively. $m_{intra}$and m_{inter} are the intra-conformer and inter ocnformer encoder embedding of the mix, respectively. r_{intra} and r_{inter} are the intra-conformer and inter-conformer encoder embedding of the ref, respectively. D_{intra} and D_{inter} denote the output of intra-decoder and inter-decoder, respectively.*

2. Methods

The architecture of Speaker Embedding Free target speaker extraction Network (SEF-Net) is shown in the Figure 2. The masking-based SEF-Net has a similar structure with Sepformer [14], which consists of audio encoder, segmentation, masking network, overlap&add, and audio decoder. We will introduce the details in this section.

2.1. Twin Audio Encoder

We apply the same method in SpEx+ [19], which uses two weight sharing audio encoders to process the mixed and reference wave separatly. These two twin audio encoders transform the time-domain inputs into an STFT-like representation:

$$E_{mix} = ReLU(conv1d(mix)), E_{mix} \in \mathbb{R}^{B\times N\times L} \tag{1}$$

$$E_{ref} = ReLU(conv1d(ref)), E_{ref} \in \mathbb{R}^{B\times N\times L} \tag{2}$$

where $mix, ref \in \mathbb{R}^{B\times T}$ denote the mixed and reference wave, respectively. The mix and ref have the same length in the proposed system. B is the batch size. N is the feature dimention and L is the number of time steps.

2.2. Segmentation

The segmentation stage splits $E_{mix}, E_{ref} \in \mathbb{R}^{B\times N\times L}$ into 3-D features $S_{mix}, S_{ref} \in \mathbb{R}^{B\times N\times K\times S}$. E_{mix} and E_{ref} are the audio encoding of mix and ref, respectively. K is the length of chunks. S is the number of chunks.

2.3. Masking Network

The architecture of the masking network is shown in the dashed box in Figure 2. The masking network applies a dual-path structure like in DPRNN [11], which consists of an Intra Module and an Inter Module. Both the Intra Module and Inter Module contain two parts: 1) two twin conformer encoders, we denote as intra- and inter-conformer. 2) A transformer decoder, we denote as intra- and inter-decoder.

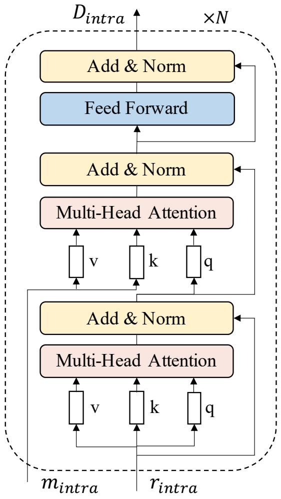

Figure 3: *The diagram of the intra-decoder. m_{intra} and r_{intra} denote the intra-conformer encoder embedding of the mix and ref, respectively. D_{intra} is the intra-decoder output.*

2.3.1. Intra Module

The Intra Module consists of two twin intra-conformer encoders and an intra-decoder. The weight shared intra-conformer encoders perform intra-chunk processing on S_{mix} and S_{ref} separately. S_{mix} is first transformed into $S'_{mix} \in \mathbb{R}^{(B*S)\times K\times N}$. Then S'_{mix} is computed in each intra-conformer layer:

$$FFN = S'_{mix} + \frac{1}{2} * FeedForward1(S'_{mix}) \tag{3}$$

$$LN = LayerNorm1(FFN) \tag{4}$$

$$MH = MultiHeadAttention(q, k, v = LN) \tag{5}$$

$$CV = (LN + MH) + Convolution(LN + MH) \tag{6}$$

$$\begin{aligned} m_{intra} = LayerNorm2(CV+ \\ \frac{1}{2} * FeedForward2(CV)) \end{aligned} \tag{7}$$

where m_{intra} denotes the intra-conformer encoding of S'_{mix}. FFN, LN, MH, CV denote the outputs of the first feed forword network, the first layernorm, the multi-head attention module in the intra-conformer, and convolution module of conformer, respectively. We process S_{ref} in the same way with S_{mix} and obtain r_{intra}. Then the m_{intra} and r_{intra} are fed into the intra-decoder.

The structure of the intra-decoder is shown in the Figure 3. We first apply multi-head attention on the r_{intra}:

$$MH' = MultiHeadAttention(q, k, v = r_{intra}) \quad (8)$$

$$LN' = LayerNorm1(MH' + r_{intra}) \quad (9)$$

where MH' denotes the output of the multi-head attention module in the intra-decoder. LN' denotes the output of the first layernorm module in the intra-decoder. Then we appply a cross multi-head attention layer which uses the LN' as the query and m_{intra} as the key and value. This cross multi-head attention module makes the intra-conformer encoder of the mixture interact well with the reference wave:

$$\begin{aligned} MH'' = MultiHeadAttention(q = LN'; \\ k, v = m_{intra}) \end{aligned} \quad (10)$$

where MH'' denote the output of the cross multi-head attenrtion module in the intra-decoder. At last, we apply a feed forword network with layernorm:

$$LN'' = LayerNorm2(MH'' + LN') \quad (11)$$

$$FFN' = FeedForward(LN'') \quad (12)$$

$$D_{intra} = LayerNorm3(FFN' + LN'') \quad (13)$$

where D_{intra} is the output of the intra-decoder.

2.3.2. Inter Module

The Inter Module, which has the same components and structure with Intra Modul in Section 2.3.1, consists of two twin inter-conformer encoders and an inter-decoder. The Inter Module performs the inter-chunk processing on the output of the Intra Module. Firstly, the D_{intra}, m_{intra}, $r_{intra} \in \mathbb{R}^{(B*S)\times K\times N}$ are tansformed into D'_{intra}, m'_{intra}, $r'_{intra} \in \mathbb{R}^{B\times N\times K\times S}$. Then the D'_{intra} is fed into the intra-conformer:

$$D'_{intra} = D'_{intra} + m'_{intra} \quad (14)$$

$$m_{inter} = F(T(D'_{intra})) \quad (15)$$

where $T(*)$ means a transformation from $D'_{intra} \in \mathbb{R}^{B\times N\times K\times S}$ to $T(D'_{intra}) \in \mathbb{R}^{(B*K)\times S\times N}$. $F(*)$ denotes the operations in the inter-conformer which are the same as Equation (3)-(7). m_{inter} is the inter-conformer encoding of D'_{intra}. Then we process the r'_{intra} in the same way with m'_{intra} and obtain r_{inter}. The inter-decoder in the Inter Module processes the m_{inter} and r_{inter} in the same way with Equation (8)-(12). The inter-decoder's output D_{inter} is the final result of the masking network.

2.4. Overlap-Add

We first transform $D_{inter} \in \mathbb{R}^{(B*K)\times S\times N}$ into $D'_{inter} \in \mathbb{R}^{B\times N\times K\times S}$ and then perform the overlap-add operation on the D'_{inter} to transform it back to a sequence:

$$M = Overlap\&Add(D'_{inter}) \quad (16)$$

where $M \in \mathbb{R}^{N\times L}$ denotes the transformed 2-D feature which is the estimation mask of the target speaker.

2.5. Audio Decoder

The audio decoder is a transposed convolution module and it has the same stride and kernel size with the encoder. The decoder takes in the product of the mask M and the audio encoding of mix and then derive the estimation of the target speaker:

$$est = T - Conv1d(M * E_{mix}), est \in \mathbb{R}^{1\times T} \quad (17)$$

3. Experiment Setup

3.1. Dataset

We simulated WSJ0-2mix-extr[1] dataset at sampling rate of 8kHz bsaed on WSJ0 corpus. The simulated dataset are divided into three sets: training set that contains 20,000 utterances of 101 speakers, development set that contains 5,000 utterances of 101 speakers, and test set that contains 3,000 utterances of 18 speakers. The speakers in training set and development set are both from WSJ0 "si_tr_s" and the utterances of two speakers are randomly selected to generate the mixed wave in relative SNR between 0 dB and 5 dB. The test set is similarly generated using the audio from WSJ0 "si_dt_05" and "si_et_05". Consistent with SpEx+ [19], the first selected speaker is chosen as the target speaker. In traing stage, we change the reference speech of the target speaker in each epoch. We cut or padded the reference speech into the same length with the mixed wave in this work.

3.2. Implementation details

The audio encoder uses an 1D convolution layer with a kernel size of 16 and a stride factor of 8. The input and output dimention of the encoder are 1 and 256, respectively. The segmentation stage splits the input into several chunks of size $K = 250$. For the conformer encoder and transformer decoder in the intra-block and inter block, we both employ 4 layers, 8 parallel attention heads, and 2048-dimensional feed-forward network. The kernel size of the conformer encoder is set to 31. The audio decoder has the same kernel size and stride factor with audio encoder. We trained our models on 4-second long segments and we used the Adam [23] as the optimizer. The initial learning rate is set to 1e-4. We train the proposed model to maximize the scale-invariant signal-to-distortion ratio (SI-SDR) [24].

4. Results and discussions

4.1. Results on WSJ0-2mix-extr

We compare the SEF-Net with other baseline speaker extraction models in terms of Signal-to-Distortion Ratio (SDR), SI-SDR and PESQ on the test set of WSJ0-2mix-extr. The results are shown in Table 1. Table 1 shows that: 1) Most target speaker extraction models require a speaker embedding whether it is from a pre-trained or a joint-learned speaker embedding extractor. 2) Our proposed SEF-Net outperforms other target speaker extraction baselines. Comparing with SpEx+, SEF-Net acchieves 3.9% and 3.4% relative improvements in terms of SDR and SI-SDR. It proves that SEF-Net performs well on the target speaker extraction task, despite has not used a speaker embedding.

[1] https://github.com/xuchenglin28/speaker_extraction

Table 1: *SDR(dB), SI-SDR(dB), and PESQ of separated speech on WSJ0-2mix-extr. "✓" and "×" denote that the model uses and not uses the speaker embedding extractor or speaker recognition loss function, respectively*

Methods	spk-embd	SDR	SI-SDR	PESQ
Mixture	-	2.60	2.50	2.31
SBM [25]	✓	9.62	9.22	2.64
SBM-C [26]	✓	11.39	10.60	2.77
TseNet [27]	✓	15.24	14.73	3.14
SpEx [18]	✓	17.15	16.68	3.36
SpEx+ [19]	✓	18.54	18.20	3.49
SEF-Net	×	**19.26**	**18.81**	**3.50**

Table 2: *SDRi(dB) and SI-SDRi(dB) of SS and SE models on the WSJ0-2mix dataset. "SS" and "SE" denote speech separation and speaker extraction, respectively. For SE, we report the average performence on the two speakers. The reference speech are chosen randomly.*

Task	Method	Parameter	SDRi	SI-SDRi
SS	DPCL [4]	13.6M	-	10.8
	uPIT [8]	92.7M	10.0	-
	DANet [6]	9.1M	-	10.5
	Chimera++ [28]	32.9M	12.0	11.5
	TasNet [29]	23.6M	13.6	13.2
	C-TasNet [10]	5.1M	15.6	15.3
	DPRNN [11]	2.6M	19.0	18.8
	Wavesplit [30]	29M	22.2	22.3
	Sepformer [14]	26M	22.3	22.4
SE	WASE [31]	7.5M	17.0	-
	SpEx [18]	10.8M	16.3	15.8
	SpEx+ [19]	11.1M	17.2	16.9
	SpExsc [32]	28.4M	18.8	18.6
SE	SEF-Net	27M	17.6	17.2

4.2. Results on WSJ0-2mix

We have also compared SEF-Net with a number of mainstream speech separation models and target speaker extraction models on WSJ0-2mix dataset. The results are shown in the Table 2. Our proposed SEF-Net outperform some SS and SE models, such as TasNet and SpEx+. However, SEF-Net is a bit short compared to the state-of-the-art models, such as Sepformer and SpExsc. There are several possible reasons: 1) This work is our initial attempt at a speaker embedding free target extraction model. There are still some details of SEF-Net that can be optimized. 2) The dynamic mixing using in Sepformer and speaker-speech cross attention in SpExsc may improve our work. Both 1) and 2) are will be our future work.

4.3. Is speaker embedding mandatory for SE tasks?

As show in Figure 2, SEF-Net obtains the target speaker information from the reference wave's conformer encoding (r_{inter}). It should be noted that r_{inter} is a shallow sequence-type feature, not a deep speaker embedding-like feature. To illustrate r_{inter} is effective for the target speaker extraction, we trained a simple speaker verification (SV) model with r_{inter} as input. This SV model consists of two linear layers. The inset accuracy results are shown in Table 3. Moreover, We visualize the average of r_{inter} and the first linear's embedding of the SV model in Figure 4. The Figure 4(A) shows that r_{inter} does not explicitly represent speaker information. The Figure 4(B) and the Table 3 show that this SV model performs well, which proves that r_{inter} does contains speaker information to some extent. Based on the above results, we conclude that adopting a speaker embedding extractor or speaker recognition loss function is not mandatory for target speaker extraction. A feature contains sufficient speaker information, such as r_{inter}, can achieve well performence on speaker extraction as well.

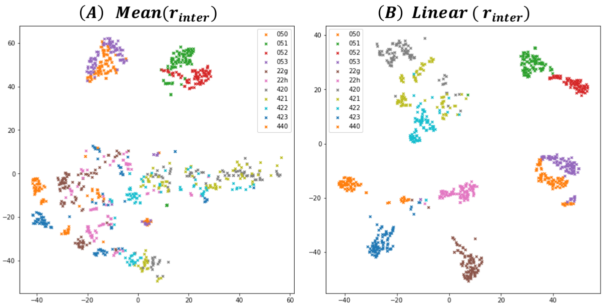

Figure 4: *t-SNE [33] of two features on 10 samples from the test set. Mean(r_{inter}) is the average of the conformer encoder output of reference wave. Linear(r_{inter}) is the first linear's embedding of the SV model.*

Table 3: *Accuracy (ACC%) results of the SV model on the training set and development (Dev) set. Mean(r_{inter}) denote the input feature is the average of r_{inter}. Linear(r_{inter}) denote the input feature is the embedding of the SV model (35 epoch).*

Feature	Train	Dev
Mean(r_{inter})	4.02	3.08
Linear(r_{inter})	93.07	89.46

5. Conclusions

This paper presents a novel approach to addressing the speaker embedding mismatch problem and showcases its practicality and potential to improve performance in target speaker extraction tasks. Specifically, we proposed SEF-Net, which is a conformer-based speaker embedding free target speaker extraction model. SEF-Net does not rely on target speaker embeddings obtained from a pre-trained or joint-learned speaker embedding extractor as other mainstream speaker extraction models. Experimental results show that our proposed model achieves comparable performance to other target speaker extraction schemes. It proves that a feature containing sufficient speaker information, not necessarily a speaker embedding, can well accomplish the target speaker extraction tasks.

6. Acknowledgements

This research is funded in part by the National Natural Science Foundation of China (62171207)Science and Technology Program of Guangzhou City (202007030011), Science and Technology Program of Suzhou City(SYC2022051) and OPPO. Many thanks for the computational resource provided by the Advanced Computing East China Sub-Center.

7. References

[1] E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *Journal of the Acoustical Society of America*, vol. 25, pp. 975–979, 1953.

[2] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[3] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE transactions on audio, speech, and language processing*, vol. 18, no. 3, pp. 550–563, 2009.

[4] Y. Z. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," in *Interspeech*, 2016.

[5] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. of ICASSP*, 2016.

[6] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Proc. of ICASSP*, 2017, pp. 246–250.

[7] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. of ICASSP*, 2017, pp. 241–245.

[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/ACM transactions on audio, speech, and language processing*, vol. 25, no. 10, pp. 1901–1913, 2017.

[9] Y. Luo and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," in *Proc. of ICASSP*, 2018, pp. 696–700.

[10] Luo, Yi and Mesgarani, Nima, "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.

[11] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *Proc. of ICASSP*, 2020.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[13] J. Chen, Q. Mao, and D. Liu, "Dual-Path Transformer Network: Direct Context-Aware Modeling for End-to-End Monaural Speech Separation," in *Proc. Interspeech*, 2020, pp. 2642–2646.

[14] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *Proc. of ICASSP*, 2021, pp. 21–25.

[15] Q. Wang, H. Muckenhirn, K. W. Wilson, P. Sridhar, Z. Wu, J. R. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. Lopez-Moreno, "Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking," in *Interspeech*, 2018.

[16] T. Li, Q. Lin, Y. Bao, and M. Li, "Atss-Net: Target Speaker Separation via Attention-Based Neural Network," in *Proc. Interspeech*, 2020, pp. 1411–1415.

[17] Z. Zhang, B. He, and Z. Zhang, "X-tasnet: Robust and accurate time-domain speaker extraction network," in *Interspeech*, 2020.

[18] 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.

[19] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "SpEx+: A Complete Time Domain Speaker Extraction Network," in *Proc. Interspeech*, 2020, pp. 1406–1410.

[20] M. Elminshawi, W. Mack, S. Chakrabarty, and E. Habets, "New insights on target speaker extraction," *ArXiv*, vol. abs/2202.00733, 2022.

[21] Z. Zhao, D. Yang, R. Gu, H. Zhang, and Y. Zou, "Target Confusion in End-to-end Speaker Extraction: Analysis and Approaches," in *Proc. Interspeech*, 2022, pp. 5333–5337.

[22] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech*, 2020, pp. 5036–5040.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[24] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr half-baked or well done?" *Proc. of ICASSP*, pp. 626–630, 2018.

[25] M. Delcroix, K. Zmolikova, K. Kinoshita, A. Ogawa, and T. Nakatani, "Single channel target speaker extraction and recognition with speaker beam," in *Proc. of ICASSP*, 2018, pp. 5554–5558.

[26] C. Xu, W. Rao, C. E. Siong, and H. Li, "Optimization of speaker extraction neural network with magnitude and temporal spectrum approximation loss," *Proc. of ICASSP*, pp. 6990–6994, 2019.

[27] Chenglin Xu and Wei Rao and Chng Eng Siong and Haizhou Li, "Time-domain speaker extraction network," *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 327–334, 2019.

[28] Z.-Q. Wang, J. L. Roux, and J. R. Hershey, "Alternative objective functions for deep clustering," *Proc. of ICASSP*, pp. 686–690, 2018.

[29] Y. Luo and N. Mesgarani, "Real-time single-channel dereverberation and separation with time-domain audio separation network," in *Proc. of Interspeech*, 2018.

[30] N. Zeghidour and D. Grangier, "Wavesplit: End-to-end speech separation by speaker clustering," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2840–2849, 2020.

[31] Y. Hao, J. Xu, P. Zhang, and B. Xu, "Wase: Learning when to attend for speaker extraction in cocktail party environments," *Proc. of ICASSP*, pp. 6104–6108, 2021.

[32] W. Wang, C. Xu, M. Ge, and H. Li, "Neural speaker extraction with speaker-speech cross-attention network," in *Interspeech*, 2021.

[33] L. van der Maaten and G. E. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.