Rethinking Singing Voice Separation With Spectral-Temporal Transformer

Shuai Yu*†, Chenxing Li*, Feng Deng* and Xiaorui Wang*
* Kuai Shou Technology Co., Beijing, China
† School of Computer Science and Technology, Fudan University, Shanghai, China
E-mail: {yushuai2021, lichenxing007}@gmail.com

***Abstract*—Recently, singing voice separation from polyphonic music using fully convolutional neural networks (CNNs) has achieved promising performance. Polyphonic music often has a long-term dependency and wide frequency bands. Therefore, a large receptive field of CNNs is critical for singing voice separation. However, the fundamental concerns still raise because of the limited receptive field of CNNs. Besides, the intrinsic spectral-temporal characteristics in the spectrum are neglected by the traditional CNNs. This paper aims to provide a rethinking of model design and proposes a pure spectral-temporal transformer-based encoder to replace CNNs as an alternative. Further, we propose an attentive fusion module to attend to spectral and temporal features and fuse them dynamically. The fused features are then fed to the decoder to obtain the final mask. Experimental results show that the proposed model outperforms several existing well-known methods.**

I. Introduction

Singing voice separation, which aims to separate singing voice from music mixture, has become an active research topic of music information retrieval (MIR). It has a lot of downstream applications of melody-based AI music, such as cover song identification [1], vocal melody extraction [2], and query-by-humming [3]. Although it is easy for a human to deal with these tasks, it is not trivial to teach machines to learn these skills.

A popular song often has two major acoustic components that are singing voice and background accompaniment. The difficulty is that there is usually a polyphonic accompaniment to the lead vocal/instrument, and this accompaniment follows the melody rhythmically and harmonically, making it difficult to separate the singing voice part. Recently, with the development of deep learning techniques, many neural network-based methods have been proposed for singing voice separation [4]–[13]. In particular, convolutional neural networks (CNNs)-based methods have attracted significant attention due to their outstanding performances. Polyphonic music often has a long-term dependency and wide frequency bands. Therefore, the large receptive field of CNNs is critical for singing voice separation. With stacked CNN layers, the receptive field of CNNs can access larger regions of the input spectrum. There are also better methods than stacked CNN layers to increase the receptive field.

One popular approach is by adopting dilated convolution [11], [12], [14], which can access a large receptive field by using dilated factors growing nonlinearly through layers. Another popular method is first to downsample the high-resolution representation to low-resolution representation and then upsample the low-resolution representation to high-resolution. It then fuses the features with a global receptive field learned from the downsampling process [7], [15], [16].

In this paper, we aim to provide a rethinking to the model design of singing voice separation. First, despite the success of dilated convolution and downsample-upsample fashion, the architecture design raises fundamental concerns that the receptive field is still limited. Second, polyphonic music often contains more than one instrument. Each instrument has its unique characteristics, resulting in their distributions in the spectrum have a certain pattern [17], [18]. However, prior CNN-based works neglect this, and the limited receptive field makes the model difficult to capture such a pattern.

Recently, transformer [19] has achieved great success in natural language processing [20], [21], computer vision [22], [23], and speech processing [24]–[28]. Despite discarding recurrence, the key point of transformer, self-attention mechanism, can draw global input-output dependencies and enables parallelization. [26] aims to model the repetition of music by adopting the self-attention mechanism. Sams-Net [28] applies a spectral transformer to music source separation and achieves performance improvement. Based on extending the idea of the transformer to this task, we propose a spectral-temporal transformer for singing voice separation. In detail, we focus on designing a spectral-temporal transformer-based encoder for singing voice separation.

To overcome the limitations above, in this paper, we replace CNNs with a pure spectral-temporal transformer. The proposed spectral-temporal transformer (STTR) can access the global receptive field and learn complex patterns better in the frequency and time axes. Concretely, we first decompose the spectrum into time sequences. We then feed the time embeddings into the temporal transformer with an embedding layer applied to the time sequences. Meanwhile, we decompose the spectrum into a frequency sequence according to the frequency bands. Another embedding layer is applied to the frequency sequence, and then the frequency embeddings are fed to the spectral transformer to learn the correlation between frequency bands. Note that we do not apply any downsampling operation but global spectral-temporal modelling at every layer of transformers. We believe that the proposed spectral-temporal transformer can offer a new perspective to this task.

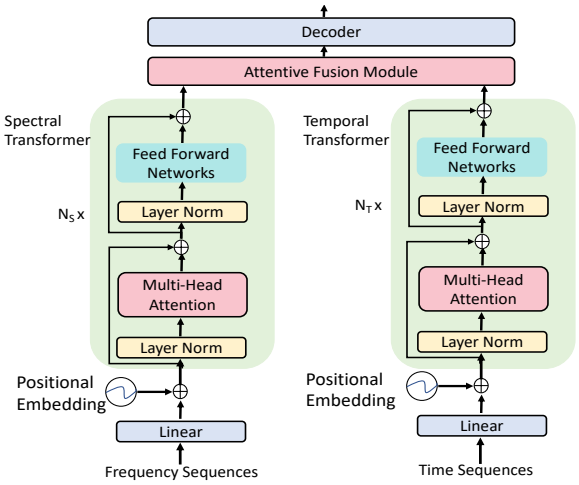


Fig. 1: The overall architecture of the proposed model. The parts highlighted with green rectangles are the proposed spectral-temporal transformer module.

We argue that simply concatenate the spectral and temporal features may hinder the further improvement of the performance. We further propose an attentive fusion module to select the features from spectral and temporal transformers dynamically. Accordingly, we fuse the features for thereafter singing voice separation. Two technique contributions are listed: i) we propose a novel spectral-temporal transformer encoder to access the global receptive field and better learn complex patterns in the frequency and time axes. There are no such works for singing voice separation in the literature to the best of our knowledge. ii) An attentive fusion module is proposed to assign weight to the spectral and temporal features dynamically.

The rest of the paper is organized as follows. In Section 2, we describe the architecture of the proposed model. In Section 3, we compare the proposed model with several previously proposed methods in singing voice separation. Finally, the conclusions and future work are given in Section 4.

II. Proposed Model

The overall architecture of our proposed spectral-temporal transformer network is shown in Fig. 1. The spectral-temporal transformer module, attentive fusion module in the proposed deep architecture is respectively addressed.

A. Model Input

We choose the short-time-Fourier-transformation (STFT) spectral magnitude as the input to the model. For computing STFT, we use a 1024-sample window size and a 512-sample hop size. To facilitate training, We split each song into 11-second segments for training and testing. As a result, each segment has 344 frames and 513 frequency bins.

B. Spectral-Temporal Transformer Module

1) Positional Embedding Layer: Unlike the positional embedding layer in the original transformer [19], to simplify the model design, we use a learning-based embedding matrix to replace the original embedding function. Given an embedding matrix $\mathbf{M}$, the embedding $\mathbf{e}$ at position i can be obtained: $\mathbf{e} = \mathbf{M}_i$, where $\mathbf{M}_i$ denotes the i-th row of $\mathbf{M}$. In this paper, we have two parallel transformers (i.e., spectral transformer and temporal transformer) leveraged for encoding the spectral and temporal sequences. Thus we use two embedding matrix $\mathbf{M}_s \in \mathbb{R}^{F\times H}$ and $\mathbf{M}_t \in \mathbb{R}^{T\times H}$, where F, T and H denote the number of frequency bins, the number of time steps, and dimensions of the embedding vector, respectively.

2) Scaled Dot-Product Attention: Self-attention, a mechanism that relates different positions of input sequences to compute representations for the inputs. Concretely, it has three inputs: queries, keys, and values. One query's output is computed as a weighted sum of the values, where each weight of the value is computed by a designed function of the query with the corresponding key. Let $\mathbf{Q} \in \mathbb{R}^{t_q\times d_q}$ be the queries, $K \in \mathbb{R}^{t_k\times d_k}$ be the keys and $V \in \mathbb{R}^{t_v\times d_v}$ be the values, where t_* are the element numbers in different inputs and d_* are the corresponding element dimensions. Normally, $t_k = t_v$, $d_q = d_k$. The outputs of self-attention is computed as:

$$\mathbf{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}, \quad (1)$$

where the scalar $1/\sqrt{d_k}$ is used to prevent softmax function into regions that have very small gradients.

3) Multi-head Attention: Multi-head attention, a core module of the spectral-temporal transformer, is applied to leveraging different attending representations jointly. Multi-head attention calculates h times scaled dot-product attention, where h means the head number. Before performing each attention, three linear projections transform the queries, keys, and values to more discriminated representations, respectively. Then, each scaled dot-product attention is calculated independently, and their outputs are concatenated and fed into another linear projection to obtain the final d_m model dimensional outputs:

$$\mathbf{MultiHead}(\mathbf{Q},\mathbf{K},\mathbf{V}) = Concat(head_1,...,head_h)\mathbf{W^O}, \quad (2)$$

$$where \quad head_i = Attention(\mathbf{Q}\mathbf{W_i^Q},\mathbf{K}\mathbf{W_i^K},\mathbf{V}\mathbf{W_i^V}). \quad (3)$$

In the equation above, since $\mathbf{Q},\mathbf{K}$, and $\mathbf{V}$ in the spectral-temporal transformer have the same dimension of d_m, the projection matrices $\mathbf{W_i^Q} \in \mathbb{R}^{d_m\times d_q}$, $\mathbf{W_i^K} \in \mathbb{R}^{d_m\times d_k}$, $\mathbf{W_i^V} \in \mathbb{R}^{d_m\times d_V}$, $\mathbf{W^O} \in \mathbb{R}^{hd_V\times d_m}$. In spectral transformer, $d_q = d_k = d_v = 513$ and in the temporal transformer $d_q = d_k = d_v = 344$. In this experiment, we set $d_m = 512$ throughout the paper.

4) Position-wise Feed Forward Network: Position-wise feed-forward network is another core module of the Spectral-Temporal Transformer module. It consists of two linear transformations with a ReLU activation in between. The dimensionality of input and output is d_m, and the inner layer has

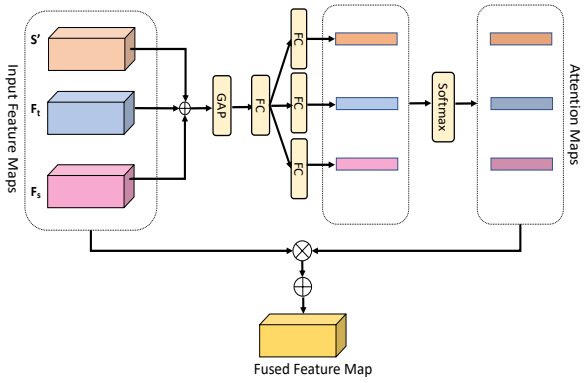


Fig. 2: The detailed architecture of the proposed attentive fusion module.

dimensionality d_{fn} . Specifically,

$$FFN(x) = max(0, \mathbf{x}\mathbf{W_1} + b_1)\mathbf{W_2} + \mathbf{b_2}, \quad (4)$$

where the weights $\mathbf{W}_1 \in \mathbb{R}^{d_m \times d_{ff}}$, $\mathbf{W}_2 \in \mathbb{R}^{d_{ff} \times d_m}$, and the biases $\mathbf{b}_1 \in \mathbb{R}^{d_{ff}}$, $\mathbf{b}_2 \in \mathbb{R}^{d_m}$. The linear transformations are the same across different positions.

C. Attentive Fusion Module

The design of the attentive fusion module (AFM) represents the second contribution of this work. Inspired by the idea of selective kernel networks [29] used in computer vision, we devise this module to dynamically attend to spectral and temporal features and fuse them. The detailed architecture is shown in Fig. 2. This module takes three inputs: the feature map $\mathbf{S}' \in \mathbb{R}^{F \times T \times C}$ is generated from the input STFT spectrum $\mathbf{S}$ via a (1×1) convolution. $\mathbf{F_s} \in \mathbb{R}^{F \times T \times C}$ is generated from the spectral transformer output via a (1×1) convolution and $\mathbf{F_t} \in \mathbb{R}^{F \times T \times C}$ is generated from the spectral transformer output via a (1×1) convolution. Firstly, an element-wise addition is performed to fuse the three inputs into a new feature map $\mathbf{\Gamma}$ and then a global average pooling (GAP) is performed to obtain global descriptor $g \in \mathbb{R}^C$.

$$g = \frac{1}{F \times T} \sum_{i<=F, j<=T} \Gamma_{ij}. \quad (5)$$

After a fully connected (FC) layer for nonlinear transformation, three FC layers are used to learn the importance of each channel of the feature maps. The softmax layer is applied to obtaining the attention map. After obtaining the attention maps, matrix multiplication is performed between the three inputs and the attention maps to obtain the weighted feature maps. Finally, we fuse the three weighted feature maps by an element-wise addition operation. The fused feature map contains rich information selected from the spectral and temporal transformers.

D. Model Architecture

The spectral-temporal transformer module aims to design a pure transformer-based encoder to replace CNNs and contribute an alternative. In particular, we use N_S-layer spectral transformer and N_T-layer temporal transformer to encode the spectral and temporal features in the encoder part. In this paper, we set $N_S = N_T = 4$ and $h = 6$ in the experiments. After obtaining features from the spectral-temporal transformer module, we propose an attentive fusion module to select the spectral and temporal features dynamically. We use two subsequent FC layers to perform the nonlinear transformation. We set the hidden dimensions to 256 and 128, respectively. We adopt a very simple decoder, and we hope that the performance gains can be easily attributed to such settings. As shown in Fig. 3, we use two branches to decode the time-frequency-based masks, vocal mask and accompaniment mask. In each branch, we use three CNN layers with kernel size (5×5), stride size (1×1), and padding size (2×2).

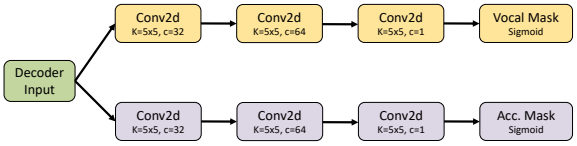


Fig. 3: The detailed architecture of the decoder of the proposed model. 'K' and 'c' stand for the kernel size and the number of channels.

III. Experiment

A. Experiment Setup

We evaluate the proposed method using the MUSDB18 dataset prepared for SiSEC 2018 [9]. In the dataset, approximately 10 hours of professionally recorded 150 songs in stereo format at 44.1kHz are available. We adopt the official split of 100 and 50 songs for training and testing, respectively. In addition, to evaluate the proposed model, we also use iKala [30], DSD100 dataset [31] to evaluate the performance. All songs are downsampled to 16 kHz. Please note that there is no overlap between training and testing sets. The quality of the separated sources are measured using the source to distortion ratio (SDR), source to interference ratio (SIR) [32].

When training the model, mean absolute error (MAE) loss is adopted between the source magnitude and the estimated magnitude. The phase of the mixture is used to restore the separated speech. We train the model on 2 NVIDIA 2080TI GPUs for a total of 100 epochs. Adam optimizer [33] is used with $\beta_1 = 0.9$, $\beta_2 = 0.98$ and $\epsilon = 1e-9$ varied the learning rate over the course of training. Meanwhile, we set both residual dropout and attention dropout to 0.1, where the residual dropout is applied to each sub-block before adding the residual information. The attention dropout is performed on the softmax activations in each attention.

B. Ablation Study

To investigate how much the proposed spectral-temporal transformer contributes to the model, we first remove the spectral transformer, and only a temporal transformer is used to encode the temporal sequence. As shown in TABLE I, the performances on both datasets are decreased. When focusing

TABLE I: Results of Ablation Study on iKala and MUSDB18 dataset. SDR and SIR values are mean values of each song. SDR/SIR(V.) denotes the SDR/SIR value on the vocal part, and SDR/SIR(A.) denotes the SDR/SIR value on the accompaniment part.

Method	iKala			
	SDR(V.)	SDR(A.)	SIR(V.)	SIR(A.)
w/o S-trans.	8.61	3.15	6.15	14.56
w/o T-trans.	6.44	3.23	9.71	4.38
w/o AFM	7.88	6.71	9.95	12.26
Proposed	**8.78**	**8.54**	**11.10**	**16.64**

(a) iKala

Method	MUSDB18			
	SDR(V.)	SDR(A.)	SIR(V.)	SIR(A.)
w/o S-trans.	0.18	15.72	4.06	11.84
w/o T-trans.	-0.29	15.44	4.54	13.26
w/o AFM	0.32	16.86	3.27	15.01
Proposed	**0.42**	**17.86**	**3.95**	**25.15**

(b) MUSDB18

on SDR on the vocal part, the performance is decreased by 1.9% on iKala and by 57.1% on MUSDB18. We can observe that MUSDB18 data is more sensitive to spectral modelling. MUSDB18 contains songs with different musical genres, which needs better spectral modelling when performing source separation. We then remove the temporal transformer and only keep the spectral transformer. The performances on both datasets are decreased by 63.1% and by 12.0%, respectively. On the contrary, the result shows iKala data is more sensitive to the temporal transformer. It indicates that popular or rock songs are more dependent on temporal modelling due to their well-organized musical structure.

We then investigate the effectiveness of the attentive fusion module. When focusing on SDR on the vocal part, the performances of the ablated version are decreased by 10.3% and 23.8% on iKala and MUSDB18, respectively. When focusing on SDR on the accompaniment part, the performances of the ablated version are decreased by 27.3% and 35.7% on iKala and MUSDB18, respectively. The results justify the assumption that direct concatenation may hinder the further improvement of the model.

C. Comparison with existing works

The performances on the three datasets are listed in TABLE II. Four commonly used and state-of-the-art methods are selected as baselines and compared in TABLE II, including fully convolution-based UNet [8], recurrent neural network-based GRU-Dilation [11], dilated convolution-based D3Net [12] and transformer-based Sams-Net [28]. We carefully tune the hyper-parameters of the baselines to ensure that they reach their peak performances on our training dataset. The proposed model and the four baseline methods are trained on the same dataset. Compared with the baseline methods, the proposed method achieves the highest score in general. The results clearly confirm the effectiveness and robustness of our proposed model. Compared with other baselines, when

TABLE II: Results of the proposed and baseline methods on iKala, MUSDB18 and DSD100 dataset. SDR and SIR values are mean values of each song. SDR/SIR(V.) denotes the SDR/SIR value on the vocal part, and SDR/SIR(A.) denotes the SDR/SIR value on the accompaniment part.

Method	iKala			
	SDR(V.)	SDR(A.)	SIR(V.)	SIR(A.)
Mixture	2.84	-4.05	2.86	-4.07
UNet [8]	8.20	-1.89	9.49	-0.80
GRU-Di. [11]	4.90	-1.31	8.46	-0.04
Sams-Net [28]	6.92	3.96	10.12	4.85
D3Net [12]	8.71	6.82	**11.66**	11.90
Proposed	**8.78**	**8.54**	11.10	**16.64**

(a) iKala

Method	MUSDB18			
	SDR(V.)	SDR(A.)	SIR(V.)	SIR(A.)
Mixture	-6.43	3.99	-6.78	3.95
UNet [8]	-0.29	12.92	1.23	16.47
GRU-Di. [11]	-2.59	14.16	-0.30	18.88
Sams-Net [28]	-0.18	15.72	**4.80**	13.55
D3Net [12]	0.09	16.57	4.29	22.68
Proposed	**0.42**	**17.86**	3.95	**25.15**

(b) MUSDB18

Method	DSD100			
	SDR(V.)	SDR(A.)	SIR(V.)	SIR(A.)
Mixture	-1.33	-4.39	-1.92	3.21
UNet [8]	2.90	11.96	3.85	15.10
GRU-Di. [11]	-0.26	12.95	1.37	16.44
Sams-Net [28]	1.93	13.68	5.03	19.95
D3Net [12]	2.24	14.92	5.44	20.75
Proposed	**3.86**	**18.00**	**6.48**	**24.94**

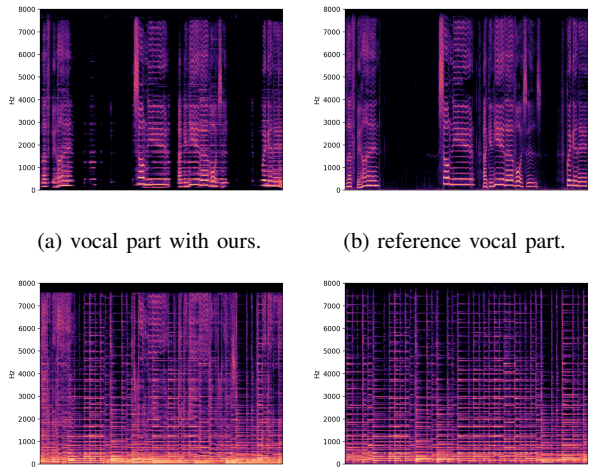
(c) DSD100

focusing on SDR on the vocal part, the proposed method outperforms the second-best D3Net by 0.8% in iKala, by 78.6% in MUSDB18 and by 72.3% in DSD100. When focusing on SDR on the accompaniment part, the proposed method outperforms the second best D3Net by 25.2% in iKala, by 7.2% in MUSDB18 and by 20.6% in DSD100.

To investigate what types of errors are solved by the proposed model, a case study is performed on a popular song: 'All Souls Moon' in the DSD100 dataset. As depicted in Fig. 4, we can observe that the proposed model generally works well on the vocal part. There are little accompaniment components in the left part of diagram (a). However, on the accompaniment part, the separated spectrogram carries lots of vocal components, which indicates that the accompaniment branch in the decoder needs to be further enhanced. We conjecture it is because the accompaniment branch learned similar parameters as the vocal branch due to the simple, fast downsampling architecture as depicted in Fig. 3. Since this paper aims to design a pure transformer-based encoder, we leave this as a research topic.

IV. Conclusion

This paper proposes a novel spectral-temporal transformer-based method to replace the conventional CNNs as an alternative for singing voice separation, which mainly contains two novel modules: spectral-temporal transformer and attentive



(a) vocal part with ours. (b) reference vocal part.

(c) acc. part with ours. (d) reference acc. part.

Fig. 4: Visualization of singing voice separation results on a popular song in the DSD100 dataset using the proposed model.

fusion. The spectral-temporal transformer is used to learn spectral and temporal features with a global receptive field. An attentive fusion module is suggested to recalibrate magnitudes and fuse the raw information for prediction. Spectral-temporal transformer and attentive fusion module are learned simultaneously in an end-to-end way. Experimental results show the proposed model outperforms several existing state-of-the-art models on three datasets. Designing a more accurate and faster method to improve singing voice separation will be our future work.

REFERENCES

[1] Joan Serra, Emilia Gómez, and Perfecto Herrera, "Audio cover song identification and similarity: background, approaches, evaluation, and beyond," in *Advances in Music Information Retrieval*, pp. 307–332. Springer, 2010.

[2] Justin Salamon, Emilia Gómez, Daniel PW Ellis, and Gaël Richard, "Melody extraction from polyphonic music signals: Approaches, applications, and challenges," *IEEE Signal Processing Magazine*, vol. 31, no. 2, pp. 118–134, 2014.

[3] Chung-Che Wang and Jyh-Shing Roger Jang, "Improving query-by-singing/humming by combining melody and lyric information," *IEEE ACM Trans. Audio Speech Lang. Process*, vol. 23, no. 4, pp. 798–806, 2015.

[4] Stefan Uhlich, Franck Giron, and Yuki Mitsufuji, "Deep neural network based instrument extraction from music," in *Proc. ICASSP*. 2015, pp. 2135–2139, IEEE.

[5] Aditya Arie Nugraha, Antoine Liutkus, and Emmanuel Vincent, "Multichannel music separation with deep neural networks," in *Proc. EUSIPCO*. 2016, pp. 1748–1752, IEEE.

[6] Naoya Takahashi and Yuki Mitsufuji, "Multi-scale multi-band densenets for audio source separation," in *Proc. WASPAA*. 2017, pp. 21–25, IEEE.

[7] Stefan Uhlich, Marcello Porcu, Franck Giron, Michael Enenkl, Thomas Kemp, Naoya Takahashi, and Yuki Mitsufuji, "Improving music source separation based on deep neural networks through data augmentation and network blending," in *Proc. ICASSP*. 2017, pp. 261–265, IEEE.

[8] Andreas Jansson, Eric J. Humphrey, Nicola Montecchio, Rachel M. Bittner, Aparna Kumar, and Tillman Weyde, "Singing voice separation with deep u-net convolutional networks," in *Proc. ISMIR*, 2017, pp. 745–751.

[9] Daniel Stoller, Sebastian Ewert, and Simon Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," in *Proc. ISMIR*, 2018, pp. 334–340.

[10] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach, "Music source separation in the waveform domain," *arXiv preprint arXiv:1911.13254*, 2019.

[11] Jen-Yu Liu and Yi-Hsuan Yang, "Dilated convolution with dilated GRU for music source separation," in *Proc. IJCAI*, 2019, pp. 4718–4724.

[12] Naoya Takahashi and Yuki Mitsufuji, "D3net: Densely connected multidilated densenet for music source separation," *Proc. ICASSP*, 2021.

[13] Weitao Yuan, Bofei Dong, Shengbei Wang, Masashi Unoki, and Wenwu Wang, "Evolving multi-resolution pooling CNN for monaural singing voice separation," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 29, pp. 807–822, 2021.

[14] Yi Luo and Nima Mesgarani, "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.

[15] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017.

[16] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. CVPR*, 2019, pp. 5693–5703.

[17] Frederick Z. Yen, Mao-Chang Huang, and Tai-Shih Chi, "A two-stage singing voice separation algorithm using spectro-temporal modulation features," in *Proc. INTERSPEECH*, 2015, pp. 3321–3324.

[18] Xiong Xiao, Engsiong Chng, and Haizhou Li, "Joint spectral and temporal normalization of features for robust recognition of noisy and reverberated speech," in *Proc. ICASSP*. 2012, pp. 4325–4328, IEEE.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Proc. NeurIPS*, 2017, pp. 5998–6008.

[20] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding.," *Proc. ICLR*, 2019.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*. 2019, pp. 4171–4186, Association for Computational Linguistics.

[22] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *CoRR*, vol. abs/2010.11929, 2020.

[23] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," *Proc. CVPR*, 2021.

[24] Linhao Dong, Shuang Xu, and Bo Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *Proc. ICASSP*. IEEE, 2018, pp. 5884–5888.

[25] Jingjing Chen, Qirong Mao, and Dong Liu, "Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation," *Proc. Interspeech 2020*, pp. 2642–2646, 2020.

[26] Yuzhou Liu, Balaji Thoshkahna, Ali Milani, and Trausti Kristjansson, "Voice and accompaniment separation in music using self-attention convolutional neural network," *arXiv preprint arXiv:2003.08954*, 2020.

[27] Sanyuan Chen, Yu Wu, Zhuo Chen, Jian Wu, Jinyu Li, Takuya Yoshioka, Chengyi Wang, Shujie Liu, and Ming Zhou, "Continuous speech separation with conformer," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5749–5753.

[28] Tingle Li, Jiawei Chen, Haowen Hou, and Ming Li, "Sams-net: A sliced attention-based neural network for music source separation," in *2021 12th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2021, pp. 1–5.

[29] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang, "Selective kernel networks," in *Proc. CVPR*, 2019, pp. 510–519.

[30] Tak-Shing Chan, Tzu-Chun Yeh, Zhe-Cheng Fan, Hung-Wei Chen, Li Su, Yi-Hsuan Yang, and Jyh-Shing Roger Jang, "Vocal activity informed singing voice separation with the ikala dataset," in *Proc. ICASSP*, 2015, pp. 718–722.

[31] Nobutaka Ono, Zafar Rafii, Daichi Kitamura, Nobutaka Ito, and Antoine Liutkus, "The 2015 signal separation evaluation campaign," in *Proc. LVA/ICA*. 2015, vol. 9237 of *Lecture Notes in Computer Science*, pp. 387–395, Springer.

[32] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Speech Audio Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[33] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, Yoshua Bengio and Yann LeCun, Eds., 2015.