Sams-Net: A Sliced Attention-based Neural Network for Music Source Separation

Tingle Li[1,3], *Jiawei Chen*[3], *Haowen Hou*[4], *Ming Li*[1,2]

[1]Data Science Research Center, Duke Kunshan University, Kunshan, China
[2]School of Computer Science, Wuhan University, Wuhan, China
[3]School of Computer Science and Technology, Tiangong University, Tianjin, China
[4]AI Platform Department, Tencent Inc., Shenzhen, China

ml442@duke.edu

Abstract

Recent studies in deep learning-based source separation have two major approaches: one approach is modeling in the spectrogram domain, and the other approach is modeling in the time domain, but all of them used pure CNN or LSTM. In this paper, we propose a Sliced Attention-based neural network (Sams-Net) at the spectrogram domain for music source separation task, which enables feature interactions from the magnitude spectrogram contribute differently to the separation. Sams-Net has two main advantages: one is that it can be easily parallel computing compared with LSTM, and the other is that it has a larger receptive field compared with CNN. Experiments indicate that our proposed Sams-Net outperforms most of the state-of-the-art methods, although it contains fewer parameters.

Index Terms: Sliced Attention, Music Source Separation, Music Information Retrieval

1. Introduction

The cocktail party effect was first proposed by Cherry [1]: How does the human brain separate a conversation from the surrounding noise. Later Bregman [2] tried to study how human brain can analyse the complex auditory signal, and proposed a framework for it. By the early 21st century, [3] attempted to reproduce or simulate the brain's ability of source separation by means of algorithms, which served as the main framework of Source Separation right now. When it comes to Music Source Separation, the first unsupervised method is [4]. Recently, the supervised methods, especially the deep learning-based methods [5, 6], have achieved great performance for this task.

Music was produced by combining individual instruments called stems. The goal of Music Source Separation is to recover those individual stems from the mixed signal [7]. In the SiSEC 2018 campaign [8], those individual stems were grouped into 4 categories: Vocals, Drums, Bass and Other. Given a song which is a mixture of these four sources, our goal is to separate it into four parts that correspond to original sources. Unlike the Speech Separation task, every single source is quite different from each other, so there is no need to use Permutation Invariant Training (PIT) [9] here. However, every song has many recurring elements [10], which may appear in the song at any time, such as individual notes, pitch, timber and chords. And this bring huge challenge for modelling.

Currently, most of the successful Music Source Separation models can be categorized into two main types, namely spectrogram based methods [5, 6, 11], and waveform based methods [12, 13, 14]. In most cases, there are generally three basic structures to bulid the deep neural networks: Feedfoward Fully Connected Network (FNN) [15], Convolutional Neural Networks (CNN) [6, 13] and Long Short-Term Memory (LSTM) [5]. And recently the CNN and LSTM have been combined to achieve state-of-the-art for Music Source Separation [16, 14]. However, all of them have certain drawbacks. On the one hand, for CNN, the receptive field may have some limitations [17]. For example, deeper CNN layers are required to obtain a larger receptive field, making training difficult. To solve this problem, a multi-scale structure was used to adapt a CNN in [13], where CNN are employed on multiple scales using downsampling like max-pooling layer, and the low resolution feature maps were upsampled layer by layer to get the original resolution. Moreover, although the pooling layer expands the receptive field and aggregates the context, it loses the spectrogram details and spatial information. On the other hand, for LSTM based networks like [11], it can not perform parallel calculations due to the time-dependent property, which makes inference taking long. Furthermore, although LSTM alleviate the problem of long-distance dependence to some extent, it still can not solve the long-term dependency problem very well, according to the recent study in [18].

The attention mechanism [19] is a recent advance in neural network modelling. It enable feature interactions contribute differently to the prediction. More importantly, the importance of the feature interaction is automatically learned from data without any human domain knowledge. It can not only bring better performance, but also insight into which feature interactions contribute more to prediction. Recently, Vaswani et al. proposed a novel neural network structure - Transformer [20], which uses only the attention mechanism structure to obtain the state-of-the-art result in the English-French translation task. Using the attention mechanism, the Transformer is a structure that can automatically capture sequence distribution. It allows the network to learn which part of the input sequence is important, and which is not. Experiments show that the Transformer is more suitable for processing sequences than LSTMs, because it can solve long-term dependency problems better than LSTMs [20]. Also, since the attention mechanism has no time-dependent limitation for calculating, the Transformer can be computed in parallel easily. Furthermore, the Transformer has a larger receptive field compared with CNNs with the same number of layers. Shortly after that, a well-known framework called BERT appeared [21], which reduced the gap between the pre-training word embedding and the downstream specific Natural Language Processing (NLP) task.

Motivated by this, we try to study the capability of attention mechanism in the source separation task, and propose a neural network for music source separation named as Sams-Net, which apply Sliced Attention mechanism to time-frequency domain. Besides, although many time-domain models have achieved better signal-to-distortion ratio (SDR) metric [22] than those of the spectrogram domain, modeling in the time domain does not produce good quality speech [23]. So we finally decide to build our model in the spectrogram domain like [11]. Experimental results show that our model has achieved a new state-of-the-art result.

2. Music Source Separation

First, the stereo music mixture $x \in \mathbb{R}^{2\times T}$ can be expressed as a linear combination of c source signals $s \in \mathbb{R}^{2\times T}$:

$$x(t) = \sum_{i=1}^{c} s_i(t) \tag{1}$$

We take the Short-time Fourier Transform (STFT) of each musical segment, so that each music segment $S_i(t, f)$ can be mapped to a 2D array of time-frequency bins $X(t, f)$:

$$X(t, f) = \sum_{i=1}^{c} S_i(t, f) \tag{2}$$

Then the reconstruction of time domain signal $\widehat{S}$ is accomplished by calculating Inverse Short-time Fourier transform (ISTFT), that is, performing element-wise multiplication between the mask M_i and the mixed spectrogram $|X(t, f)|$, then multiplied by the imaginary phase power of e.

$$\widehat{S}_i(t, f) = (|X(t, f)| \odot M_i(t, f)) \times e^{\angle(X(t,f))i} \tag{3}$$

where $\angle(X(t, f))$ is the phase of mixed musical segment.

So this task can be denoted as minimizing the following objective function:

$$\mathcal{L} = \arg\min_{M} \sum_{i=1}^{c} \left\| S_i(t, f) - \widehat{S}_i(t, f) \right\|_2^2 \tag{4}$$

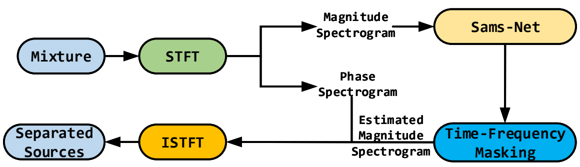

Figure 1: *The flow chart of our training system.*

3. Proposal: Sams-Net

In this section, we first introduce what Scale Dot-Product Attention is, which serves as the core of neural attention network. Then, we introduce the Sliced Attention, which slice the spectrogram in advance instead of directly perform Scaled Dot-Product Attention. The structure of Sams-Net can be seen in figure 2.

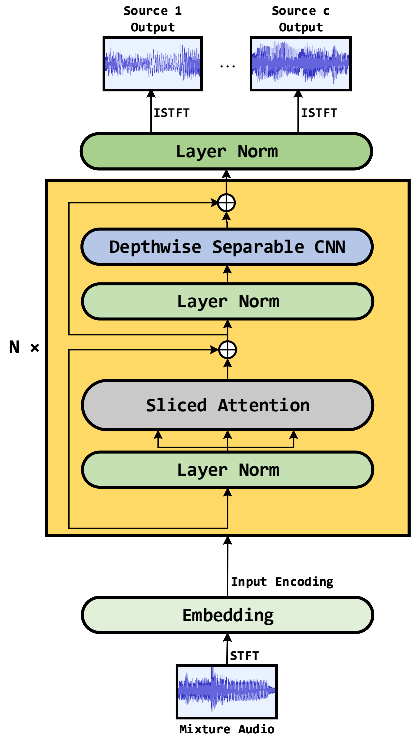

Figure 2: *Model architecture of Sams-Net.*

3.1. Scaled Dot-Product Attention

The attention mechanism is a method of learning the distribution of its weight by calculating the similarity of sequences' elements. The commonly used methods of attention mechanism are dot-product, concatenation, perceptron, etc. Here we use dot-product. In our task, after we embed the magnitude of the spectrogram into the d_k dimensional space using a CNN layer with the kernel size 3×3, we use a 1×1 CNN layer for three times, to get the query Q, key K, and value V respectively. Then we compute the dot products of the query Q with key K, divide each by $\sqrt{d_k}$, where d_k is the number of the feature maps. Finally, we apply a softmax function to obtain the weights on the values V. The formula is as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

where $1/\sqrt{d_k}$ plays a regulating role, so that the inner product is not too large and avoid gradient vanishing problem.

3.2. Multi-Head Attention

In order to extract more information from magnitude spectrogram, we use a well-known attention mechanism called Multi-Head Attention here [20]. We first perform Scaled Dot-Product Attention to calculate attention value. Then repeat this operation for h times, and concatenate all the result we get following the channel axis. Finally, performing the convolution to get the final attention values of the same shapes as the original.

$$\begin{aligned} MultiHead(Q, K, V) = Concat(&head_1, head_2, \\ &\cdots, head_h) \cdot W^O \end{aligned} \tag{6}$$

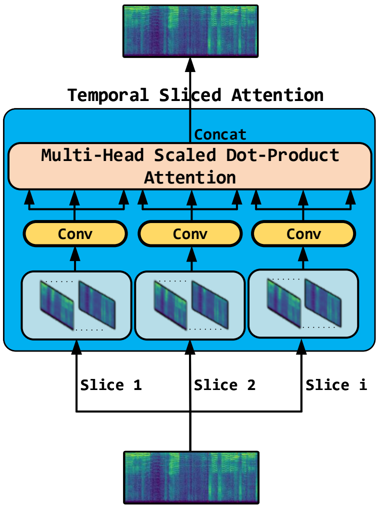

Figure 3: *Model architecture of Sliced Attention module.*

where $head_i = Attention(Q \cdot {W_i}^Q, K \cdot {W_i}^K, V \cdot {W_i}^V)$, and W^Q, W^K, W^V is the CNN layer with the kernel size 1×1. W^O is a 3×3 CNN layer, which is used for compressing the number of the feature maps.

3.3. Sliced Attention

In the music source separation task, since the duration of every song is very long, it is better to slice the time-frame of the magnitude spectrogram in advance, instead of directly perform Scaled Dot-Product Attention. Because music source separation is not like machine translation, it does not require too long-term dependencies, but require more useful sample values to be interacted to predict the current sample value. So we proposed a new method called Sliced Attention. First, we slice the time-frame of the input magnitude spectrogram Q, K, V into equal i groups respectively, and each slice retains complete frequency information. Meanwhile, the scope of attention is narrowed down so that the slices are more concerned with the features that are most likely to affect each other's recent time period. Then each group (Q_i, K_i, V_i) is paired one by one to perform Multi-Head Attention. For example, Q_1, K_1 and V_1 is paired to compute the attention value, and so on, till the end of the spectrogram. Finally, i attention values are concatenated as the output SA. It can be expressed as follows:

$$\begin{cases} Q = \{Q_1, Q_2, \cdots, Q_i\} \\ K = \{K_1, K_2, \cdots, K_i\} \\ V = \{V_1, V_2, \cdots, V_i\} \end{cases} \tag{7}$$

$$slice_i = MultiHead(Q_i, K_i, V_i) \tag{8}$$

$$\text{SA} = Concat(slice_1, slice_2, \cdots, slice_i) \tag{9}$$

3.4. Depthwise Separable CNN

Considering the size of our model, we choose Depthwise Separable CNN [24] rather than conventional CNN. This architecture is widely known for [25]. With this architecture, the size of model parameters can be greatly reduced, and the prediction accuracy of the model is less affected, compared with the conventional CNN. Specifically, it has two steps: Depth-wise CNN and Point-wise CNN. For Depth-wise CNN, the feature map is performed convoluted independently with the kernel size 3×3. Then go to the Point-wise CNN step, that is, passing through a CNN layer with the kernel size 1×1. Their mathematical formulation is as follows (we use $\odot$ to denote the element-wise product):

$$DepthwiseConv(W, SA)_{(i,j)} = \sum_{k,l}^{K,L} W_{(k,l)} \odot SA_{(i+k,j+l)} \tag{10}$$

$$PointwiseConv(W, SA)_{(i,j)} = \sum_{m}^{d_k} W_m \cdot SA_{(i,j,m)} \tag{11}$$

where W is the kernel of the CNN layer, y is the output of Sliced Attention, d_k denotes the number of the feature maps, and K, L denote the length and width of magnitude spectrogram, respectively.

3.5. Layer Norm

We use residual connections in the two sub-layers above, and then perform layer normalization [26]. So the output of each sublayer is:

$$LayerNorm(x + Sublayer(x)) \tag{12}$$

where $Sublayer(x)$ is a function implemented by the sublayer itself.

4. Experiment and Results

4.1. Experimental setup

We evaluated Sams-Net on the MUSDB18 dataset [27], which is prepared for the SiSEC 2018 campaign [8]. MUSDB18 has 100 and 50 songs in the training and test set, respectively. In this dataset, each song contains a mixture and its four sources, Vocals, Bass, Drums, and Other, all of which are recorded in stereo format with the sampling rate of 44.1 kHz. During the training stage, we randomly select 6 seconds duration from each training track as the training data. While in the validation stage, 1 complete track is randomly selected from 14 tracks that [27] proposed as the validation data.

The window function of the Short-time Fourier transform (STFT) is set to the Hamming window, and the frame length and hop size are set to 4096 and 1024, respectively. Also, data augmentation [28] is used as inputs. For the evaluation on MUSDB18, we used the museval package [8] and BSSEval v4 toolbox [22] for a fair comparison with previously reported results. The SDR values are the average of the median SDR metric of each song.

The PyTorch framework [33] was used to build our models, and we train them with 2 NVIDIA TITAN RTX GPU. The networks were trained to estimate the magnitude spectrogram by minimizing the Mean Square Error (MSE) with the Adam optimizer [34], with a learning rate of 0.0001. If the validation loss does not descend after 140 epoch, the early stop will be performed and the training stage will end. Because of the large use of GPU memory, we can only train 3 Sliced Attention modules with 2 heads and 64 channel dimensions.

4.2. Results and Discussions

As shown in table 2, we find Slice Attention (slice=2, 4, 8, 12) is better than traditional Scaled Dot-Product Attention (slice=1).

Table 1: *A comparison SDR metric of our proposed method with other models on the test set of MUSDB18 dataset.*

Model	**Domain**	**# Param**	**Test SDR (dB)**				
			Vocals	Drums	Bass	Other	Average
IRM oracle	N/A	N/A	9.43	8.45	7.12	7.85	8.21
DeepConvSep [29]	Spectrogram	0.32M	2.37	3.14	0.17	-2.13	0.89
WaveNet [30]	Waveform	3.30M	3.35	4.13	2.49	2.60	2.60
Wave-U-Net [13]	Waveform	10.20M	3.25	4.22	3.21	2.25	3.23
Spect U-Net [31]	Spectrogram	9.84M	5.74	4.66	3.67	3.40	4.37
Open-Unmix [11]	Spectrogram	8.90M	6.32	5.73	5.23	4.02	5.36
Demucs [14]	Waveform	66.42M	6.29	6.08	**5.83**	4.12	5.58
Meta-TasNet [32]	Waveform	12.00M	6.40	5.91	5.58	**4.19**	5.52
MMDenseLSTM [16]	Spectrogram	4.88M	6.60	6.41	5.16	4.15	5.58
Sams-Net	Spectrogram	3.70M	**6.61**	**6.63**	5.25	4.09	**5.65**

Specifically, when the number of slices increases from 1 to 12, the SDR metric of four sources increases gradually. However, when the slices further increase from 12 to 24, the performance drops gradually. We guess it is because the magnitude segments are too small, resulting in the loss of phonetic information. For the MUSDB18 dataset, the optimal slice number is around 12, but that doesn't apply to all the data. In our opinion, the optimal slices number for Slice Attention may be determined by unsupervised methods like Spectral Clustering [35], without knowing the number of clusters on the dataset.

Why Sliced Attention is better? In our view, unlike the machine translation task, in the music source separation task, the interdependence of each song is not that stable, because it often has a sudden change in style. For example, playing with drums at this moment, but suddenly change to bass or a pure human voice at the next moment. Under this circumstance, if we focus on the whole magnitude spectrogram, it may attend to irrelevant points, resulting in a lack of focus on the key points.

As shown in table 1, we also compare against previously published and state-of-the-art models for the MUSDB18 dataset (some models trained with extra data are not listed here). The SDR metric was either taken from the SiSEC 2018 [8] evaluation scores [13, 16, 31] or from the related papers [11, 29, 14, 30, 32], and we show SDR metric with the median over frames, median over tracks here. The Ideal Ratio Mask oracle (IRM oracle) [11] is the topline of the spectrogram based method, which computes the best possible mask using the ground truth sources. It is worth mentioning that the

Table 2: *A comparison SDR metric of Sams-Net among several slice numbers.*

Slice	**Vocals**	**Drums**	**Bass**	**Other**	**Average**
1	6.42	6.55	5.21	3.97	5.53
2	6.44	6.59	**5.32**	4.04	5.60
4	6.42	6.57	5.29	4.08	5.59
8	6.61	6.56	5.23	4.06	5.62
12	**6.61**	**6.63**	5.25	**4.09**	**5.65**
16	6.51	6.45	5.19	4.01	5.54
18	6.57	6.37	4.94	3.88	5.44
24	6.26	5.98	4.55	3.68	5.12

current baseline for the spectrogram domain models is Open-Unmix [11], and our Sams-Net has surpassed it a lot. Besides, the state-of-the-art models are [14] and [16], and from the table we can see that our Sams-Net achieves new best scores on Vocals and Drums categories, and the score on Average category is better than both of them as well.

Moreover, the model parameters are calculated by the code[1] if they are not listed in their respective papers. As shown in the table 1, the parameters of Sams-Net has about 3.7M, which is smaller than most of the previous methods. Specifically, compared with the baseline Open-Unmix [11], whose parameters are about 8.9M, our model achieves better SDR results. The small number of parameters in our model brings great advantages, such as lower memory usage and less computation complexity. Besides, we believe further increase the parameters of Sams-Net can achieve even better performance. But due to the lack of GPU, we can't train the bigger model.

5. Conclusions

In this paper, we propose a Sliced Attention-based neural network for music source separation, named Sams-Net, which improve Music Source Separation performance by discriminating the importance of different feature interactions. We also propose a new attention mechanism named Sliced Attention, which learns the importance of each feature interaction from every segment of magnitude spectrogram via neural attention network. As shown by the experiment results above, our Sams-Net achieved better SDR performance compared with other methods, although the parameters of which are much less than most of them.

For future works, in order not to ignore the difference of the phase between the mixed and separated audio, we intend to model in the time-domain or use the Griffin-Lim algorithm [36] to reconstruct the phase signals. Also, we hope to apply the unsupervised method to our model, so that the model can automatically search out the optimal number of slice.

[1] https://discuss.pytorch.org/t/how-do-i-check-the-number-of-parameters-of-a-model/

6. Acknowledgements

This research is funded in part by the National Natural Science Foundation of China (61773413), Key Research and Development Program of Jiangsu Province (BE2019054), Six talent peaks project in Jiangsu Province (JY-074), Guangzhou Municipal People's Livelihood Science and Technology Plan (201903010040).

7. References

[1] 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.

[2] A. S. Bregman, *Auditory scene analysis: The perceptual organization of sound*. MIT press, 1994.

[3] N. Roman and D. Wang, "Pitch-based monaural segregation of reverberant speech," *The Journal of the Acoustical Society of America*, vol. 120, no. 1, pp. 458–469, 2006.

[4] Y. Li and D. Wang, "Separation of singing voice from music accompaniment for monaural recordings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1475–1487, 2007.

[5] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Singing-voice separation from monaural recordings using deep recurrent neural networks." in *ISMIR*, 2014, pp. 477–482.

[6] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep u-net convolutional networks," in *ISMIR*, 2017, pp. 745–751.

[7] E. Vincent, "Musical source separation using time-frequency source priors," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 91–98, 2005.

[8] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2018, pp. 293–305.

[9] 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.

[10] J. Paulus, M. Müller, and A. Klapuri, "State of the art report: Audio-based music structure analysis." in *ISMIR*. Utrecht, 2010, pp. 625–636.

[11] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-unmix-a reference implementation for music source separation," *Journal of Open Source Software*, 2019.

[12] A. 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," *arXiv preprint arXiv:1609.03499*, 2016.

[13] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," *arXiv preprint arXiv:1806.03185*, 2018.

[14] A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Music Source Separation in the Waveform Domain," HAL, Tech. Rep. 02379796v1, 2019.

[15] A. J. Simpson, G. Roma, and M. D. Plumbley, "Deep karaoke: Extracting vocals from musical mixtures using a convolutional deep neural network," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2015, pp. 429–436.

[16] N. Takahashi, N. Goswami, and Y. Mitsufuji, "Mmdenselstm: An efficient combination of convolutional and recurrent neural networks for audio source separation," in *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 106–110.

[17] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Advances in neural information processing systems*, 2016, pp. 4898–4906.

[18] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," *A field guide to dynamical recurrent neural networks. IEEE Press*, 2001.

[19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[21] 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.

[22] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[23] A. Pandey and D. Wang, "A new framework for supervised speech enhancement in the time domain." in *Interspeech*, 2018, pp. 1136–1140.

[24] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[25] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[26] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[27] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "Musdb18-a corpus for music separation," HAL, Tech. Rep. 02190845v1, 2017.

[28] S. Uhlich, M. Porcu, F. Giron, M. Enenkl, T. Kemp, N. Takahashi, and Y. Mitsufuji, "Improving music source separation based on deep neural networks through data augmentation and network blending," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 261–265.

[29] P. Chandna, M. Miron, J. Janer, and E. Gómez, "Monoaural audio source separation using deep convolutional neural networks," in *International conference on latent variable analysis and signal separation*. Springer, 2017, pp. 258–266.

[30] F. Lluís, J. Pons, and X. Serra, "End-to-end music source separation: is it possible in the waveform domain?" *arXiv preprint arXiv:1810.12187*, 2018.

[31] J.-Y. Liu and Y.-H. Yang, "Denoising auto-encoder with recurrent skip connections and residual regression for music source separation," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018, pp. 773–778.

[32] D. Samuel, A. Ganeshan, and J. Naradowsky, "Meta-learning extractors for music source separation," *arXiv preprint arXiv:2002.07016*, 2020.

[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.

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

[35] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[36] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.