STEREOPHONIC MUSIC SOURCE SEPARATION WITH SPATIALLY-INFORMED BRIDGING BAND-SPLIT NETWORK

Yichen Yang[1,2]*, Haowen Li*[1]*, Xianrui Wang*[1,2]*, Wen Zhang*[1]*, Shoji Makino*[2]*, and Jingdong Chen*[1]

[1]Center of Intelligent Acoustics and Immersive Communications,
Northwestern Polytechnical University, Xi'an, China
[2]Graduate School of Information, Production and Systems,
Waseda University, Kitakyushu, Japan

ABSTRACT

Stereophonic music source separation (MSS) is a problem of extracting individual source tracks, e.g. bass, drums, vocals, from a stereo music recording. Deep neural network (DNN) based MSS systems have demonstrated great promise though spatial panning cues and time-frequency spectral structures in stereo music have not yet been fully explored in such systems and methods. This paper presents a spatially-informed MSS method using a bridging band-split neural network that incorporates both spatial and spectral information. The spatial panning angles of each target source are used as input of the network, along with the time-frequency spectrograms. Moreover, the inter-track correlations are exploited for further performance improvement. Experiments show that the proposed method outperforms significantly the baseline systems as the result of using spatial cues, spectral characteristics, and inter-track relationships.

***Index Terms*—**Stereophonic music source separation, bridging band-split network, spatial information

1. INTRODUCTION

Music source separation, a crucial task in music information retrieval (MIR), has various applications such as transcription [1], music remixing [2–4], and music education [5], to name but a few. Existing MSS methods can be broadly categorized into two classes: signal processing based and deep neural network (DNN) based methods. The former uses the statistical properties between different sources to separate the signals while the latter leverages the representation learning capabilities of deep networks to separate the mixture with supervised training.

With the rapid development of DNN and the associated technologies and resources, DNN-based approaches have become increasingly popular. In earlier studies, DNNs were used to estimate either ideal ratio masks (IRMs) [6–11] or complex IRMs [12–16] to reconstruct the signal in the time-frequency domain or the feature domain, which has demonstrated the state-of-the-art performance for MSS [17–26]. However, these methods still suffer from a number of major limitations, including but not limited to: 1) the spatial information is an inherent property of stereophonic music as different instruments have different panning angles, but such information is neglected in the existing methods; 2) the time-frequency spectral structure, another important property of stereophonic music, is in general not fully exploited; 3) DNN models are generally trained independently for different tracks while the inter-track information is neglected. All the aforementioned information, if properly used, should greatly improve the separation performance.

To utilize inter-track information, based on the Open-Unmix (UMX) [27], the CrossNet-UMX (X-UMX) [28] was proposed. X-UMX incorporates a *CrossNet* with a bridging structure to extract shared information between different network branches for different output tracks. More recently, spatial cues were integrated into X-UMX, based on which the so-called SpaInNet, a spatially-informed stereophonic MSS system was developed [29]. By using source panning angles as the *a priori* knowledge, this method is able to further improve separation performance though spectral characteristics of the music signals are still neglected in such system.

To explore the spectral structures for MSS, a method with band-split recurrent neural network (BSRNN) was proposed [30–32], which splits the full time-frequency spectrum into subbands tailored for vocals and accompaniment, in which a dual-path RNN (DPRNN), similar to the architecture in [33], extracts features from the subbands and time frames to separate vocals or instruments from a mono input. However, BSRNN trains models independently for each output track, ignoring the inter-track information. As shown in [28], leveraging mutual information between tracks is an effective approach to enhancing separation performance.

In this work, we propose a spatially-informed CrossNet multi-channel BSRNN system, which offers the following augmentations. 1) It extends BSRNN to handle multi-channel inputs by merging features across channels in the band-split stage; 2) It incorporates a CrossNet to share representations between network branches, thereby exploiting the inter-track information; 3) It integrates spatial cues in the form of source panning angles to further improve separation. Together, these augmentations enable the modeling of both spectral characteristics and inter-track relationships within a multi-channel framework to leverage stereophonic information. Simulation results demonstrate that the proposed system is able to achieve significant performance gain in comparison with the baseline approaches.

2. SIGNAL MODEL AND PROBLEM FORMULATION

Consider a music signal consists of signal components from K sources. The output stereophonic signals $\mathbf{x}(t,f) \in \mathbb{C}^{M\times 1}$ ($M = 2$ is the number of channels) in the short-time Fourier transform

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2021ZD0201502 and in part by the NSFC Grants: 62171373, 61831019, 62192713 and 62271401, and in part by the China Scholarship Council (CSC).

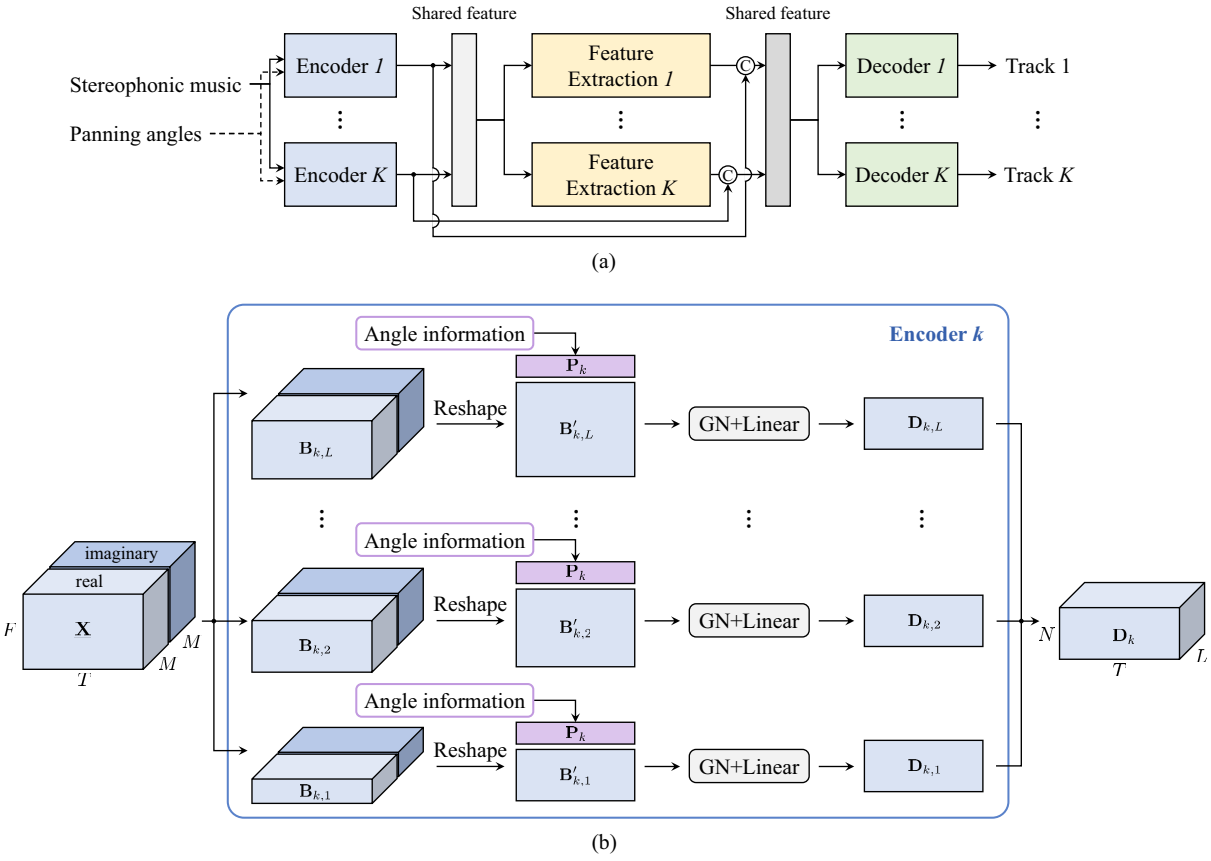

Fig. 1: (a) Pipeline of the proposed SpaIn-X-MBSRNN. (b) Architecture of the k-th encoder corresponding to the k-th source, where the symbol "C" represents concatenation operation.

(STFT) domain can be expressed as

$$\mathbf{x}(t,f) = \sum_{k=1}^{K} \mathbf{x}_k(t,f), \tag{1}$$

$$\mathbf{x}_k(t,f) = [X_{k,\mathrm{L}}(t,f) \quad X_{k,\mathrm{R}}(t,f)]^T, \quad k = 1,2,\ldots,K, \tag{2}$$

where f and t denote the frequency-bin and time-frame indices, respectively, $\mathbf{x}_k(t,f)$ is signal from the k-th source, which could be the vocal or one of the instruments, $X_{k,\mathrm{L}}(t,f)$ and $X_{k,\mathrm{R}}(t,f)$ are the left and the right channels of the k-th source, and the superscript $(\cdot)^T$ denotes the transpose operator. By incorporating spatial information, such as the panning angles, into the system, the DNN-based MSS approach can be mathematically formulated as

$$\mathbf{y}(t,f) = \mathrm{DNN}\left[\mathbf{x}(t,f), \boldsymbol{\phi}\right], \tag{3}$$

where $\mathrm{DNN}[\cdot]$ represents the DNN-based separation system, $\boldsymbol{\phi} = [\phi_1 \quad \phi_2 \quad \cdots \quad \phi_K]^T \in \mathbb{C}^{K\times 1}$ is the vector of the panning angles of K sources, and $\mathbf{y}(t,f) \in \mathbb{C}^{K\times 1}$ is the K separated sources corresponding to the K angles.

3. SPAIN-X-MBSRNN ARCHITECTURE

The proposed architecture, as shown in Fig. 1(a), consists of three main components: a multi-channel band-split encoder that incorporates spatial embeddings, a dual-path RNN for feature extraction, and a mask estimation decoder for source separation.

3.1. Encoder

3.1.1. Spatial Embeddings

To integrate spatial information, this work adopts the spatial embeddings approach from SpaInNet [29]. Specifically, the panning angle $-45^\circ \le \phi_k \le +45^\circ$ is for the source k is used to generate the spatial embeddings $\mathbf{p}_k \in \mathbb{R}^{I\times 1}$ to represent the stereo position, where I is the dimension of the spatial embedding. If $0^\circ \le \phi_k \le +45^\circ$, $\mathbf{p}_k$ is calculated as

$$\mathbf{p}_k(2i) = \sin\left(\frac{\phi_k}{45^{\frac{2i}{I}}}\right), \quad \mathbf{p}_k(2i+1) = \cos\left(\frac{\phi_k}{45^{\frac{2i}{I}}}\right), \tag{4}$$

for $i = 1,2,\ldots,I/2$. For the negative angles, i.e., $-45^\circ \le \phi_k < 0^\circ$, the spatial embedding is calculated as

$$\mathbf{p}_k(2i) = \sin\left(\frac{\phi_k}{45^{\frac{I-2i}{I}}}\right), \quad \mathbf{p}_k(2i+1) = \cos\left(\frac{\phi_k}{45^{\frac{I-2i}{I}}}\right). \tag{5}$$

Let us assume that the source location is fixed over time. Then, $\mathbf{p}_k$ can be extended along the time frame axis to form the full spatial embedding $\mathbf{P}_k \in \mathbb{R}^{I\times T}$ where each column equals $\mathbf{p}_k$.

3.1.2. Multi-Channel Band-Split with Spatial Embedding

The encoder adopts the band-split scheme from [30] to handle the multi-channel inputs by incorporating spatial embeddings for every source. As shown in Fig. 1(b), the encoder input $\underline{\mathbf{X}} \in \mathbb{R}^{F\times T\times 2M}$,

which consists of M channels, can be expressed as

$$\underline{\mathbf{X}} = \{\underline{\mathbf{X}}(t)\}_{t=1}^{T}, \tag{6}$$

$$\underline{\mathbf{X}}(t) = [\underline{\mathbf{x}}(t,1) \quad \underline{\mathbf{x}}(t,2) \quad \cdots \quad \underline{\mathbf{x}}(t,F)]^T, \tag{7}$$

$$\underline{\mathbf{x}}(t,f) = \left[\mathbf{x}_{\text{re}}^T(t,f) \quad \mathbf{x}_{\text{im}}^T(t,f)\right]^T, \tag{8}$$

where $\underline{\mathbf{X}}(t) \in \mathbb{R}^{F\times 2M}$ is the input feature in the t-th time frame, $\underline{\mathbf{x}}(t,f) \in \mathbb{R}^{2M\times 1}$ is the input vector at the (t,f) bin, and $\mathbf{x}_{\text{re}}(t,f)$ and $\mathbf{x}_{\text{im}}(t,f)$ are the real and imaginary part of $\mathbf{x}(t,f)$, respectively. Without loss of generality, let us assume that the full band of the k-th source is split into a total of L subbands with the bandwidth of the l-th band being $Q_{k,l}$. The input feature can then be rewritten as

$$\underline{\mathbf{X}}(t) \triangleq \left[\mathbf{B}_{k,1}^T(t) \quad \mathbf{B}_{k,2}^T(t) \quad \cdots \quad \mathbf{B}_{k,L}^T(t)\right]^T, \tag{9}$$

where $\mathbf{B}_{k,l}(t) \in \mathbb{R}^{Q_{k,l}\times 2M}$ is the l-th subband spectrograms of $\underline{\mathbf{X}}(t)$ for the k-th source and $\sum_l Q_{k,l} = F$. Note that the bandwidth $Q_{k,l}$ can vary for different sources. Stacking $\mathbf{B}_{k,l}(t)$ into a vector, adding the time axis to form $\mathbf{B}'_{k,l} \in \mathbb{R}^{2MQ_{k,l}\times T}$, and concatenating with the spatial embedding to form the input feature of the encoder, one can calculate the following output:

$$\mathbf{D}_k = \{\mathbf{D}_{k,l}\}_{l=1}^{L}, \tag{10}$$

$$\mathbf{D}_{k,l} = \text{Linear}_{k,l}\left[\text{GroupNorm}\left([(\mathbf{B}'_{k,l})^T\ \mathbf{P}_k^T]^T\right)\right], \tag{11}$$

where $\mathbf{D}_k \in \mathbb{R}^{N\times L\times T}$ is the output feature of the k-th encoder, $\mathbf{D}_{k,l} \in \mathbb{R}^{N\times T}$ is the subband feature corresponding to the l-th subband, N is the feature dimension, and $\text{GroupNorm}(\cdot)$ and $\text{Linear}(\cdot)$ denote, respectively, the group normalization layer and the fully connected linear layer.

3.1.3. CrossNet through Bridging Architecture

To investigate shared features across various source branches, a bridging architecture is incorporated prior to the feature extraction and mask estimation modules, similar to [28]. Specifically, before feature extraction, the first shared feature on the left of Fig. 1(a) is generated through the bridging structure as

$$\mathbf{D}' = \frac{1}{K}\sum_{k=1}^{K}\mathbf{D}_k, \tag{12}$$

where $\mathbf{D}' \in \mathbb{R}^{N\times L\times T}$ denotes the first shared feature for feature extraction. Note that in order to form $\mathbf{D}'$ between the K sources, the feature dimension N and the number of the subbands L must be the same for all sources. By connecting the output features from different encoders corresponding to separate sources, the relationship between these sources can be jointly analyzed using the shared feature.

3.2. Feature extraction

With the shared feature $\mathbf{D}'$, the DPRNN from [30] is incorporated. This module consists of K parallel branches with identical structures to extract features for K sources. Each branch employs S repeating layers with two long short-term memory (LSTM) structures modeling sequence- and band-level features across dimensions T and L, respectively. This process can be summarized as

$$\mathbf{F}_{k,s} = \text{Linear}_{k,s,1}\left\{\text{LSTM}_{k,s,1}\left[\text{GroupNorm}\left(\mathbf{D}'\right)\right]\right\}, \tag{13}$$

$$\mathbf{F}'_{k,s} = \text{Linear}_{k,s,2}\left\{\text{LSTM}_{k,s,2}\left[\text{GroupNorm}\left(\mathbf{F}_{k,s}\right)\right]\right\}, \tag{14}$$

where $\text{LSTM}_{k,s,1}(\cdot)$ and $\text{LSTM}_{k,s,2}(\cdot)$ represent two LSTM structures in the sth repeated layer where $s = 1, 2, \ldots, S$, and $\mathbf{F}'_{k,S} \in \mathbb{R}^{N\times L\times T}$ is the output of the feature extraction module. The second bridging architecture generates the second shared feature for the mask estimation module, i.e.,

$$\mathbf{D}'' = \{\mathbf{D}''_l\}_{l=1}^{L}, \tag{15}$$

$$\mathbf{D}''_l = \frac{1}{K}\sum_{k=1}^{K}[(\mathbf{D}_{k,l}^T\ (\mathbf{F}'_{k,S,l})^T]^T, \tag{16}$$

where $\mathbf{D}'' \in \mathbb{R}^{2N\times L\times T}$ is the second shared feature and $\mathbf{F}'_{k,S,l} \in \mathbb{R}^{N\times T}$ is the l-th matrix of the $\mathbf{F}'_{k,S}$ for $l = 1, 2, \ldots, L$.

3.3. Decoder

With the second shared feature as the input, the mask for the k-th source is estimated as

$$\mathbf{G}_k = \text{Decoder}\left(\mathbf{D}''\right), \tag{17}$$

where $\mathbf{G}_k \in \mathbb{C}^{F\times T}$ is the complex mask for the k-th source, and $\text{Decoder}(\cdot)$ represents the decoder containing group normalization followed by two fully connected linear layers with a hyperbolic tangent (tanh) and a gated linear unit (GLU) as the activation function. The estimated source can finally be obtained as

$$\mathbf{Y}_k = \mathbf{G}_k \circ \mathbf{X}_{\text{L}}, \tag{18}$$

where $\circ$ represents the Hadamard product (i.e., the element-wise multiplication), $\mathbf{X}_{\text{L}} = \{\sum_k X_{k,\text{L}}\}_{f=1,t=1}^{F,T}$ is the left-channel mixture music signal, and $\mathbf{Y}_k \in \mathbb{C}^{F\times T}$ is the estimated k-th source. Note that in this work the left channel is chosen as the reference channel for source separation.

3.4. Training Objective

The multi-domain loss (MDL) [28] is used to optimize the proposed method in both the STFT and time domains.

A mean squared error (MSE) loss is used between the estimated and ground truth power spectrograms in the STFT domain as

$$\mathcal{L}_{\text{MSE}} = \frac{1}{KTF}\sum_{k=1}^{K}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[|X_{k,\text{L}}(t,f)| - |Y_k(t,f)|\right]^2, \tag{19}$$

where $Y_k(t,f)$ is the (t,f)-th element of $\mathbf{Y}_k$. For the time-domain loss, an additional inverse STFT (iSTFT) is applied to both $X_{k,\text{L}}(t,f)$ and $Y_k(t,f)$. The scale-invariant signal-to-distortion ratio (SI-SDR) [34] between the time-domain source signal $x_{k,\text{L}}(n)$ and estimated signal $y_k(n)$ is then calculated, i.e.,

$$\mathcal{L}_{\text{SI-SDR}} = -\frac{1}{K}\sum_{k=1}^{K}10\log_{10}\left[\frac{\mathbb{E}\left(|\alpha\mathbf{x}_{k,\text{L}}|^2\right)}{\mathbb{E}\left(|\alpha\mathbf{x}_{k,\text{L}} - \mathbf{y}_k|^2\right)}\right], \tag{20}$$

where $\mathbf{x}_{k,\text{L}} = [x_{k,\text{L}}(1) \quad x_{k,\text{L}}(2) \quad \cdots \quad x_{k,\text{L}}(N)]^T$ is the time-domain source signal vector, $\mathbf{y}_k = [y_k(1) \quad y_k(2) \quad \cdots \quad y_k(N)]^T$ is the estimated source vector, $\alpha = \mathbf{y}_k^T\mathbf{x}_{k,\text{L}}/\|\mathbf{x}_{k,\text{L}}\|^2$ is the scaling factor, and $\mathbb{E}(\cdot)$ denotes the expectation operation. The overall multi-domain cost function is

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda\mathcal{L}_{\text{SI-SDR}}, \tag{21}$$

where λ is the weighting parameter for the multi-domain loss.

4. EVALUATION

4.1. Experimental settings

To generate training data that simulates stereo signals, we use the constant power panning (CPP) law to randomly assign panning

Models	Vocals		Drum		Bass		Other		All	
	uSDR	cSDR	uSDR	cSDR	uSDR	cSDR	uSDR	cSDR	uSDR	cSDR
Mixture	-10.02	-9.76	-4.73	-4.65	-8.83	-8.65	-8.69	-8.54	-8.07	-7.90
UMX [27]	4.21	3.63	5.13	5.17	3.86	4.04	2.23	2.19	3.86	3.76
X-UMX [28]	4.31	4.29	5.30	5.32	4.14	4.26	2.98	2.94	4.18	4.20
SpaIn-X-UMX [29]	5.07	4.77	5.96	5.92	5.03	4.90	3.67	3.59	4.93	4.80
MBSRNN	11.41	12.03	13.54	13.84	9.19	9.46	7.93	8.89	10.52	11.06
X-MBSRNN	11.31	11.97	13.08	13.22	11.03	11.52	9.12	9.67	11.14	11.60
SpaIn-X-MBSRNN	11.73	11.32	13.84	13.84	12.34	12.50	11.79	12.14	12.43	12.45

Table 1: The separation performance of the baseline and proposed methods on the spatialized MUSDB18-HQ dataset.

Frequency Band (kHz)	< 1	1-4	4-8	8-16	16-20
Vocals/Other (Hz)	100	250	500	1k	2k
Drum/Bass (Hz)	50	250	1k	2k	-

Table 2: Bandwidth for frequency bands.

angles to each vocal or accompaniment source signals. For the k-th source signal $x_k(n)$, the left and right channels are defined through CPP as follows:

$$x_{k,\mathrm{L}}(n) = (\sqrt{2}/2)(\cos\phi_k + \sin\phi_k)x_k(n), \quad (22)$$

$$x_{k,\mathrm{R}}(n) = (\sqrt{2}/2)(\cos\phi_k - \sin\phi_k)x_k(n), \quad (23)$$

where $x_{k,\mathrm{L}}(n)$ and $x_{k,\mathrm{R}}(n)$ are the left and right channel signals of the simulated stereo signals corresponding to the k-th source, respectively. All source signals are taken from the MUSDB18-HQ dataset [35] with a sampling rate of 44.1kHz. For training and validation, every audio track is partitioned into 3 second segments. The panning angle for each source was randomly selected, with a minimum separation of 10° between adjacent sources.

Data augmentation has been shown to play a critical role in improving the effectiveness of MSS [36]. In this work, we utilized an energy-based sound activity detection method to remove silent segments during training, similar to [30]. The remaining active segments are randomly mixed to increase diversity in the dataset. We also apply random energy rescaling within ± 5 dB and a random dropout mechanism (with probability 0.05) to simulate inactive sources.

The spatial embedding dimension I and feature dimension N in the encoder are both set to 128. Similar to [30], for simplicity, two bandwidth division schemes are used in this work, as shown in Table 2. The remaining frequency range is treated as a single subband. With the band-split schemes described above, the total number of subbands for all sources is 41. For the feature extraction module, the number of repeated stacks S is set to 3.

The test set of the MUSDB18-HQ database is used to evaluate the baseline and proposed systems. After applying the same panning process, the stereophonic music signals are partitioned into segments with a segment length of 3 seconds and an overlapping of 0.5 seconds. Data augmentation is not used at this stage.

To evaluate the separation performance of the proposed systems, the utterance-level SDR (uSDR) [37] and chunk-level SDR (cSDR) [38] metrics are used. Note that proposed MBSRNN is a multi-channel extension of the BSRNN in [30], the proposed X-MBSRNN is the MBSRNN with a bridging structure through CrossNet, and the proposed SpaIn-X-MBSRNN is the spatially-informed X-MBSRNN. The extraction performances of the presented algorithms are compared to those of Open-Unmix (UMX) [27], CrossNet-UMX (X-UMX) [28], and spatially-informed X-UMX (SpaIn-X-UMX) [29].

4.2. Results and discussion

The separation performance of the presented algorithms and baselines for vocals and other instruments are shown in Table 1. Compared to the three baseline systems, the proposed system with band-split encoders and DPRNN feature extraction achieved clear improvements in separation performance. This shows that a well-designed subband scheme tailored to different spectral patterns is beneficial.

In comparison with MBSRNN, X-MBSRNN achieved noticeable performance improvement in separating bass and other instruments though its separation performance for vocals and drums is slightly worse. The overall performance is better, which shows that incorporating inter-tracks information helps improve separation performance. SpaIn-X-MBSRNN is shown to be the most effective algorithm and its overall uSDR and cSDR reach, respectively, 12.43 dB and 12.45 dB, which are much higher than those of all other studied methods. As seen, the performance difference between SpaIn-X-MBSRNN and X-MBSRNN is more significant for the category marked as "Other". The underlying reason, we believe, is because this category includes various instruments such as piano and violin. In such case, incorporating spatial information is more useful for improving performance.

5. CONCLUSION

In this paper, we presented a new method for stereophonic music source separation. The inter-track information between different sources is firstly explored through the bridging architecture in the multi-channel band-split network. Further, by incorporating spatial information into the bridging multi-channel band-split network, the proposed method can efficiently separate vocals and instruments using the panning angle of every source as the *a priori* information. Simulation results showed that the proposed method outperformed several state-of-the-art algorithms, which demonstrated the benefits of utilizing spatial cues and tailored subband processing for stereophonic music source separation.

6. REFERENCES

[1] E. Benetos, S. Dixon, Z. Duan, and S. Ewert, "Automatic music transcription: An overview," *IEEE Signal Process. Mag.*, vol. 36, no. 1, pp. 20–30, Jan. 2019.

[2] O. Gillet and G. Richard, "Extraction and remixing of drum tracks from polyphonic music signals" in *Proc. IEEE WASPAA*, 2005, pp. 315–318.

[3] J. Woodruff, B. Pardo, and R. B. Dannenberg, "Remixing stereo music with score-informed source separation" in *Proc. ISMIR*, 2006, pp. 314–319.

[4] J. Pons, J. Janer, T. Rode, and W. Nogueira, "Remixing music using source separation algorithms to improve the musical experience of cochlear implant users," *J. Acoust. Soc. Am.*, vol. 140, no. 6, pp. 4338–4349, Dec. 2016.

[5] E. Cano, G. Schuller, and C. Dittmar, "Pitch-informed solo and accompaniment separation towards its use in music education applications," *EURASIP J. Adva. Signal Process.*, vol. 2014, no. 23, pp. 1–19, Feb. 2014.

[6] A. Narayanan and D. L. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition" in *Proc. IEEE ICASSP*, 2013, pp. 7092–7096.

[7] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proc. IEEE ICASSP*, 2014, pp. 1581–1585.

[8] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. IEEE ICASSP*, 2016, pp. 31–35.

[9] A. Li, M. Yuan, C. Zheng, and X. Li, "Speech enhancement using progressive learning-based convolutional recurrent neural network," *Appl. Acoust.*, vol. 166, pp. 107347, Sept. 2020.

[10] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, Aug. 2019.

[11] X. Wang, N. Pan, J. Benesty, and J. Chen, "On multiple-input/binaural-output antiphasic speaker signal extraction," in *Proc. IEEE ICASSP*, 2023, pp. 1–5.

[12] D. S. Williamson, Y. Wang, and D. L. Wang, "Complex ratio masking for monaural speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 3, pp. 483–483, Mar. 2016.

[13] D. S. Williamson, and D. L. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 7, pp. 1492–1501, Jul. 2017.

[14] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex U-Net" in *Proc. ICLR*, 2018.

[15] S. Park, T. Kim, K. Lee, and N. Kwak, "Music source separation using stacked hourglass networks," *arXiv preprint arXiv:1805.08559*, 2018.

[16] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, "DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement" in *Proc. Interspeech*, 2020, pp. 2472–2476.

[17] E. Cano, D. FitzGerald, A. Liutkus, M. D. Plumbley, and F.-R. Stöter, "Musical source separation: An introduction," *IEEE Signal Process. Mag.*, vol. 36, no. 1, pp. 31–40, Jan. 2019.

[18] E. M. Grais, G. Roma, A. J. R. Simpson, and M. Plumbley, "Combining mask estimates for single channel audio source separation using deep neural networks" in *Proc. Interspeech*, 2016, pp. 3339–3343.

[19] P. Seetharaman, G. Wichern, S. Venkataramani, and J. Le Roux, "Class-conditional embeddings for music source separation," in *Proc. IEEE ICASSP*, 2019, pp. 301–305.

[20] V. S. Kadandale, J. F. Montesinos, G. Haro, and E. Gómez, "Multi-channel U-Net for music source separation" in *Proc. IEEE MMSP*, 2020, pp. 1–6.

[21] Q. Kong, Y. Cao, H. Liu, K. Choi, and Y. Wang, "Decoupling magnitude and phase estimation with deep ResUNet for music source separation," *arXiv preprint arXiv:2109.05418*, 2021.

[22] H. Liu, Q. Kong, and J. Liu, "CWS-PResUNet: Music source separation with channel-wise subband phase-aware ResUNet," *arXiv preprint arXiv:2112.04685*, 2021.

[23] M. Kim, W. Choi, J. Chung, D. Lee, and S. Jung, "KUIELab-MDX-Net: A two-stream neural network for music demixing," *arXiv preprint arXiv:2111.12203*, 2021.

[24] L. Chen, X. Zheng, C. Zhang, L. Guo, and B. Yu, "Multi-scale temporal-frequency attention for music source separation" in *Proc. IEEE ICME*, 2022, pp. 1–6.

[25] A. Défossez, "Hybrid spectrogram and waveform source separation," *arXiv preprint arXiv:2111.03600*, 2021.

[26] S. Rouard, F. Massa, and A. Défossez, "Hybrid transformers for music source separation" in *Proc. IEEE ICASSP*, 2023, pp. 1–5.

[27] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-Unmix - A reference implementation for music source separation," *J. Open Source Softw.*, vol. 4, no. 41, pp. 1667, Sept. 2019.

[28] R. Sawata, S. Uhlich, S. Takahashi, and Y. Mitsufuji, "All for one and one for all: Improving music separation by bridging networks" in *Proc. IEEE ICASSP*, 2021, pp. 51–55.

[29] D. Petermann and M. Kim, "SpaIn-Net: Spatially-informed stereophonic music source separation" in *Proc. IEEE ICASSP*, 2022, pp. 106–110.

[30] Y. Luo and J. Yu, "Music source separation with band-split RNN," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 1893–1901, May 2023.

[31] J. Yu and Y. Luo, "Efficient monaural speech enhancement with universal sample rate band-split RNN" in *Proc. IEEE ICASSP*, 2023, pp. 1–5.

[32] J. Yu, Y. Luo, H. Chen, R. Gu, and C. Weng, "High fidelity speech enhancement with band-split RNN," *arXiv preprint arXiv:2212.00406*, 2022.

[33] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation" in *Proc. IEEE ICASSP*, 2020, pp. 46–50.

[34] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR Half-baked or Well Done?" in *Proc. IEEE ICASSP*, 2019, pp. 626–630.

[35] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "MUSDB18-HQ - an uncompressed version of MUSDB18," Aug. 2019, doi: 10.5281/zenodo.3338373.

[36] L. Prétet, R. Hennequin, J. Royo-Letelier, and A. Vaglio, "Singing voice separation: A study on training data" in *Proc. IEEE ICASSP*, 2019, pp. 506–510.

[37] Y. Mitsufuji, G. Fabbro, S. Uhlich, F.-R. Stöter, A. Défossez, M. Kim, W. Choi, C.-Y. Yu, and K.-W. Cheuk "Music demixing challenge 2021," *Front. Signal Process.*, vol. 1, pp. 18, Jan. 2022.

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