Neural Fast Full-Rank Spatial Covariance Analysis for Blind Source Separation

Yoshiaki Bando*†, Yoshiki Masuyama*‡, Aditya Arie Nugraha†, and Kazuyoshi Yoshii†§
*National Institute of Advanced Industrial Science and Technology, Japan
†Center for Advanced Intelligence Project (AIP), RIKEN, Tokyo, Japan
‡Department of Computer Science, Tokyo Metropolitan University, Japan
§Graduate School of Informatics, Kyoto University, Japan

***Abstract*—This paper describes an efficient unsupervised learning method for a neural source separation model that utilizes a probabilistic generative model of observed multichannel mixtures proposed for blind source separation (BSS). For this purpose, amortized variational inference (AVI) has been used for directly solving the inverse problem of BSS with full-rank spatial covariance analysis (FCA). Although this unsupervised technique called neural FCA is in principle free from the domain mismatch problem, it is computationally demanding due to the full rankness of the spatial model in exchange for robustness against relatively short reverberations. To reduce the model complexity without sacrificing performance, we propose neural FastFCA based on the jointly-diagonalizable yet full-rank spatial model. Our neural separation model introduced for AVI alternately performs neural network blocks and single steps of an efficient iterative algorithm called iterative source steering. This alternating architecture enables the separation model to quickly separate the mixture spectrogram by leveraging both the deep neural network and the multichannel optimization algorithm. The training objective with AVI is derived to maximize the marginalized likelihood of the observed mixtures. The experiment using mixture signals of two to four sound sources shows that neural FastFCA outperforms conventional BSS methods and reduces the computational time to about 2 % of that for the neural FCA.**

***Index Terms*—blind source separation, amortized inference, joint-diagonalization, neural source separation**

I. Introduction

Sound source separation forms the basis of various machine listening systems including distant speech recognition [1]–[3] and sound event detection [4], [5]. Neural source separation has achieved excellent performance thanks to the expression power of deep neural networks (DNNs) trained with a large number of pairs of mixture signals and their corresponding source signals [6]–[8]. However, such supervised training suffers from domain mismatch and a lack of source signals in target environments. As a promising alternative, blind source separation (BSS) [9]–[11] has thus been investigated to work with little prior information about the sources and microphones.

Modern BSS methods are based on probabilistic generative models of multichannel mixture signals [9]–[12]. Such a probabilistic model consists of a source model representing the power spectral densities (PSDs) of the sources and a spatial model representing the spatial covariance matrices (SCMs) of the sources.

This work was supported in part by the JST ACT-X under Grant JPMJAX200N and NEDO.

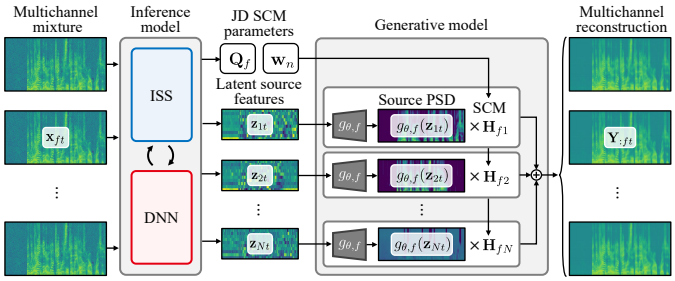

Fig. 1: The overview of the proposed neural FastFCA.

Multichannel non-negative matrix factorization (MNMF) [10], for example, is based on an NMF-based source model assuming the low-rankness of the PSDs and a full-rank spatial model assuming the full-rankness of the SCMs. While the full-rank SCMs can deal with small source movements and reverberation, their estimation is often unstable and requires an expensive computational cost due to their too-high degrees of freedom. FastMNMF [11], [13] mitigates this problem by assuming the source SCMs to be jointly-diagonalizable (JD). The JD SCM is also full-rank but is represented by a weighted sum of rank-1 SCMs common to all the sources. This constraint is reported to efficiently reduce the computational cost and improve the separation performance compared to the original MNMF [11].

To represent complex structures of source spectra, neural source models using variational autoencoders (VAEs) [14] have been proposed [15]–[18]. For example, a multichannel VAE (MVAE) [17] replaces the source model of MNMF with the decoder of a VAE pre-trained on isolated source signals. This source model can also be trained only with mixture signals by neural full-rank spatial covariance analysis (FCA) [18]. Neural FCA trains the source generative model (decoder) by introducing an inference (encoder) model that estimates latent features of the source model from a multichannel mixture. The decoder and encoder models are jointly trained to maximize the likelihood of the MVAE for training data of multichannel mixtures. The neural FCA was reported to perform on par with the supervised MVAE for speech separation [18].

In this paper, we propose a BSS method called neural FastFCA based on the integration of the JD spatial model and the neural source model (Fig. 1). The original neural FCA estimates the full-rank SCMs by an expectation-maximization (EM) al-

gorithm, which requires a high computational cost. In contrast, we assume the JD spatial model and extend the inference model to estimate the JD SCMs quickly in the network. Specifically, we introduce a network building block that diagonalizes an observed mixture by an efficient algorithm called iterative source steering (ISS) [19], [20]. We alternately stack the ISS-based diagonalization blocks and DNN blocks such that the intermediate diagonalization (quasi-separation) results can be used to estimate the latent source features. The networks are jointly trained to separate unseen mixture signals in an unsupervised manner.

The main contribution of this study is to integrate the state-of-the-art BSS techniques of the JD spatial model [11], the neural source model [18], and the ISS-based inference model [19]. This combination enables the proposed method to train the inference (separation) model and the neural source model in an unsupervised manner to achieve high separation performance and a small computational cost. The experimental results with simulated mixture signals of two to four speech sources demonstrate that our blind method outperforms conventional BSS methods. In addition, our method reduces the computational cost to about 2 % of that for the original neural FCA.

II. Background

This section first briefly overviews the existing BSS methods and then introduces a neural BSS method called neural FCA.

A. Blind source separation

BSS methods typically assume that an M-channel mixture signal $\mathbf{x}_{ft} \in \mathbb{C}^M$ is a sum of N source signals $s_{nft} \in \mathbb{C}$:

$$\mathbf{x}_{ft} = \sum_{n=1}^{N} \mathbf{a}_{nf} s_{nft}, \tag{1}$$

where $t = 1, \ldots, T$ and $f = 1, \ldots, F$ are time and frequency indices, respectively, and $\mathbf{a}_{nf} \in \mathbb{C}^M$ is the steering vector for source n. Each source signal s_{nft} is then assumed to follow a zero-mean complex Gaussian distribution as follows:

$$s_{nft} \sim \mathcal{N}_{\mathbb{C}}\left(0, \lambda_{nft}\right), \tag{2}$$

where $\lambda_{nft} \in \mathbb{R}_+$ represents the PSD of source n. By marginalizing the source signal s_{nft}, the following multivariate Gaussian likelihood is obtained:

$$\mathbf{x}_{ft} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}, \sum_{n=1}^{N} \lambda_{nft} \mathbf{H}_{nf}\right), \tag{3}$$

where $\mathbf{H}_{nf} = \mathbf{a}_{nf}\mathbf{a}_{nf}^{\mathsf{H}} \in \mathbb{S}_+^{M\times M}$ is an SCM for source n at frequency f. BSS is performed by estimating the λ_{nft} and $\mathbf{H}_{nf}$ to maximize this likelihood with sufficient assumptions to effectively restrict the model's redundant flexibility. Independent low-rank matrix analysis (ILRMA) [21], for example, assumes λ_{nft} to be low-rank for solving frequency permutation ambiguity. MNMF [12] replaces $\mathbf{H}_{nf}$ with a full-rank SCM for allowing small source movements and reverberations.

The computational cost for estimating the full-rank SCMs can be efficiently reduced by using the JD SCMs [11], [13]. Specifically, this formulation represents SCMs $\mathbf{H}_{nf}$ by a diagonalizer $\mathbf{Q}_f \in \mathbb{C}^{M\times M}$ common for all the sources and diagonal elements $\mathbf{w}_n \in \mathbb{R}_+^M$ for each source as follows:

$$\mathbf{H}_{nf} = \mathbf{Q}_f^{-1}\mathrm{diag}(\mathbf{w}_n)\mathbf{Q}_f^{-\mathsf{H}}. \tag{4}$$

The diagonalizer $\mathbf{Q}_f$ is optimized to maximize Eq. (3) with an iterative projection [9] or ISS [22] algorithm, and $\mathbf{w}_n$ is optimized with a multiplicative update rule [11]. FastMNMF combines this JD spatial model with a low-rank source model and has been reported to perform better than MNMF while working at a similar computational cost to ILRMA [11].

B. Neural full-rank spatial covariance analysis

A powerful way to represent source signals is to utilize a DNN that can precisely capture their complex spectra [15]–[17]. The deep spectral model [15]–[17] assumes that the PSD λ_{nft} is generated by a latent source feature $\mathbf{z}_{nt} \in \mathbb{R}^D$ and a non-linear function (*i.e.*, DNN) $g_{\theta,f} : \mathbb{R}^D \to \mathbb{R}_+$ as follows:

$$\lambda_{nft} = g_{\theta,f}(\mathbf{z}_{nt}), \tag{5}$$

where θ is a set of the network parameters of $g_{\theta,f}$. The latent source feature $\mathbf{z}_{nt}$ is typically assumed to follow the standard Gaussian distribution:

$$\mathbf{z}_{nt} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right), \tag{6}$$

and is supposed to represent the features of source spectra, such as pitches and envelopes.

The neural FCA [18] trains the neural source model $g_{\theta,f}$ as a decoder of a VAE by introducing an inference (encoder) model with network parameters ϕ to estimate the posterior distribution $q_\phi(\mathbf{z}_{nt} \mid \mathbf{X})$ from an observed mixture $\mathbf{X} \triangleq \{\mathbf{x}_{ft}\}_{f,t=1}^{F,T}$. Let $\mathbf{Z} \triangleq \{\mathbf{z}_{nt}\}_{n,t=1}^{N,T}$ and $\mathbf{H} \triangleq \{\mathbf{H}_{nf}\}_{f,n=1}^{F,N}$ be the sets of latent features and SCMs, respectively. This method assumes the generative model of multichannel mixtures with Eqs. (3), (5), and (6). Based on this generative model, the encoder and decoder are jointly trained in an unsupervised manner to maximize the following evidence lower bound (ELBO):

$$\mathcal{L} = \mathbb{E}_{q_\phi}[\log p_\theta(\mathbf{X} \mid \mathbf{Z}, \mathbf{H})] - \mathcal{D}_{\mathrm{KL}}[q_\phi(\mathbf{Z} \mid \mathbf{X}) \mid p(\mathbf{Z})], \tag{7}$$

where $\mathbb{E}_{q_\phi}[\cdot]$ and $\mathcal{D}_{\mathrm{KL}}[\cdot \mid \cdot]$ are the expectation by the posterior q_ϕ and the Kullback-Leibler (KL) divergence, respectively. The network parameters θ and ϕ are optimized by stochastic gradient ascent [23], and the SCMs $\mathbf{H}_{nf}$ are optimized at each network update with an EM algorithm [24]. The maximization of the ELBO corresponds to the maximization of the log-marginal likelihood $p(\mathbf{X} \mid \mathbf{H})$, and can be considered as BSS performed for the training mixture signals.

Once the networks are optimized, they are used to separate unseen mixture signals. Neural FCA has been reported to perform better than existing BSS methods including FastMNMF and on par with the supervised MVAE in speech separation [18]. This method, however, requires a high computational cost for estimating the SCMs. In addition, the separation performance is limited because the inference network does not utilize the intermediate separation results, which are usually utilized in the conventional BSS methods during their iterative algorithms.

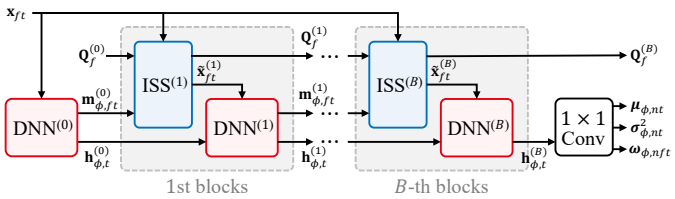

Fig. 2: The block diagram of the inference model.

III. Neural Fast Full-Rank Spatial Covariance Analysis

We extend the original neural FCA with the JD spatial model to reduce the computational cost without sacrificing performance. In addition, we introduce an inference model based on the tight integration of ISS-based blocks and DNN blocks for quickly separating multichannel mixture signals.

A. Generative model of multichannel mixture signals

Our method called neural FastFCA is based on the JD spatial model of Eqs. (3) and (4) and the neural source model of Eqs. (5) and (6). The resulting generative model is as follows:

$$\mathbf{x}_{ft} \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}, \mathbf{Q}_f^{-1}\left\{\sum_{n=1}^{N} g_{\theta,f}(\mathbf{z}_{nt})\mathrm{diag}(\mathbf{w}_n)\right\}\mathbf{Q}_f^{-\mathsf{H}}\right). \quad (8)$$

B. Inference model

The inference (separation) model of neural FastFCA estimates SCM parameters $\mathbf{Q} \triangleq \{\mathbf{Q}_f\}_{f=1}^F$ and $\mathbf{W} \triangleq \{\mathbf{w}_n\}_{n=1}^N$ as well as the posterior distribution $q_\phi(\mathbf{Z} \mid \mathbf{X})$ from an observed mixture $\mathbf{X}$. We utilize both the DNNs and multichannel optimization techniques [19] for quick source separation. Specifically, as shown in Fig. 2, the inference network estimates the parameters by alternately performing $B+1$ DNN blocks and B ISS blocks within the network. The b-th ISS block [19] updates the diagonalizer $\mathbf{Q}_f \triangleq [\mathbf{q}_{f1}, \dots \mathbf{q}_{fM}]^{\mathsf{H}}$ by iterating the following ISS update rule for $m = 1, \dots, M$:

$$\mathbf{Q}_f \leftarrow \mathbf{Q}_f - [v_{fm1}, \dots, v_{fmM}]^{\mathsf{T}}\mathbf{q}_{fm}^{\mathsf{H}}, \quad (9)$$

$$v_{fmm'} = \begin{cases} \frac{\mathbf{q}_{fm'}^{\mathsf{H}}\mathbf{U}_{fm'}\mathbf{q}_{fm}}{\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{U}_{fm'}\mathbf{q}_{fm}} & (\text{if } m' \neq m) \\ 1 - (\mathbf{q}_{fm}^{\mathsf{H}}\mathbf{U}_{fm}\mathbf{q}_{fm})^{-\frac{1}{2}} & (\text{if } m' = m) \end{cases} \quad (10)$$

where $\mathbf{U}_{fm} \in \mathbb{S}_+^{M\times M}$ is an auxiliary SCM. This SCM is calculated by the mixture $\mathbf{x}_{ft}$ and a TF mask $m_{\phi,ftm'}^{(b-1)} \in [0,1]$ predicted by the $(b-1)$-th DNN block:

$$\mathbf{U}_{fm'} = \frac{1}{T}\sum_{t=1}^{T} m_{\phi,ftm'}^{(b-1)} \cdot \mathbf{x}_{ft}\mathbf{x}_{ft}^{\mathsf{H}}. \quad (11)$$

Let $\mathbf{Q}_f^{(b)}$ be the output of the b-th ISS block. This update rule converges to the maximum likelihood estimate of $\mathbf{Q}_f$ by using an appropriate mask $m_{\phi,ftm'}^{(b)}$ and initial value $\mathbf{Q}_f^{(0)}$ [22]. The b-th DNN block, on the other hand, outputs the TF mask $m_{\phi,ftm'}^{(b)}$ and an internal feature $\mathbf{h}_{\phi,t}^{(b)}$ $(t = 1, \dots, T)$ passed to the next block. The input of the DNN block is an intermediate diagonalized (quasi-separated) spectrogram $\tilde{\mathbf{x}}_{ft}^{(b)} \triangleq \mathbf{Q}_f^{(b)}\mathbf{x}_{ft} \in \mathbb{C}^M$ concatenated with internal features $\mathbf{h}_{\phi,t}^{(b-1)}$.

After performing B ISS blocks and $B+1$ DNN blocks, the last internal feature $\mathbf{h}_{\phi,t}^{(B)}$ is converted to $q_\phi(\mathbf{Z} \mid \mathbf{X})$ and $\mathbf{W}$ with an output (1×1-convolution) layer. The posterior distribution $q_\phi(\mathbf{Z} \mid \mathbf{X})$ is estimated as the following Gaussian distribution:

$$q_\phi(\mathbf{Z} \mid \mathbf{X}) \leftarrow \prod_{n,t,d=1}^{N,T,D} \mathcal{N}\left(z_{ntd} \mid \mu_{\phi,ntd}, \sigma_{\phi,ntd}^2\right), \quad (12)$$

where $\mu_{\phi,ntd} \in \mathbb{R}$ and $\sigma_{\phi,ntd}^2 \in \mathbb{R}_+$ are the network outputs representing the mean and variance of $\mathbf{Z}$, respectively. On the other hand, the diagonal elements $\mathbf{w}_n$ are estimated as a normalized average of frequency-wise estimates $\mathbf{w}'_{\phi,fn} \in \mathbb{R}_+^M$:

$$\mathbf{w}_n \leftarrow \frac{1}{F}\sum_{f=1}^{F} \frac{1}{\frac{1}{M}\|\mathbf{w}'_{\phi,fn}\|_1}\mathbf{w}'_{\phi,fn} \quad (13)$$

$$\mathbf{w}'_{\phi,fn} = \sum_{t=1}^{T} \boldsymbol{\omega}_{\phi,nft} \circ \left|\tilde{\mathbf{x}}_{ft}^{(B)}\right|^{\circ 2} \quad (14)$$

where $\boldsymbol{\omega}_{\phi,nft} \in [0,1]^M$ is a network output to represent an M-channel TF mask, and $\circ$ and $|\cdot|^{\circ 2}$ indicate the element-wise product and element-wise absolute square, respectively.

C. Amortized variational inference

The generative and inference models are trained by using only multichannel mixture signals as an amortized variational inference [14]. As in the original neural FCA, the training objective for each mixture signal is an ELBO as follows:

$$\mathcal{L} = \mathbb{E}_{q_\phi}[\log p_\theta(\mathbf{X} \mid \mathbf{Z}, \mathbf{W}, \mathbf{Q})] - \mathcal{D}_{\mathrm{KL}}[q_\phi(\mathbf{Z} \mid \mathbf{X}) \mid p(\mathbf{Z})],$$

where $\mathbf{Q}$ denotes the network output $\mathbf{Q}^{(B)}$ for simplicity. The KL term can be calculated in the same way as in [18] and used to solve the frequency permutation ambiguity. The first term of the ELBO, on the other hand, is calculated approximately from the inference results $\mathbf{Q}$, $\mathbf{W}$, and $q_\phi(\mathbf{Z} \mid \mathbf{X})$ as follows:

$$\mathbb{E}[\log p_\theta(\mathbf{X} \mid \mathbf{Z}, \mathbf{W}, \mathbf{Q})] \approx T\sum_{f=1}^{F} \log\left|\mathbf{Q}_f\mathbf{Q}_f^{\mathsf{H}}\right| - \sum_{f,t,m=1}^{F,T,M}\left\{\log\tilde{y}_{ftm} + \frac{|\tilde{x}_{ftm}|^2}{\tilde{y}_{ftm}}\right\}, \quad (15)$$

where $\tilde{\mathbf{x}}_{ft} \triangleq [\tilde{x}_{ft1}, \dots, \tilde{x}_{ftM}]^T = \mathbf{Q}_f\mathbf{x}_{ft} \in \mathbb{C}^M$ is the diagonalized observation, and $\tilde{y}_{ftm} \triangleq \sum_{n=1}^{N} w_{nm}g_{\theta,f}(\mathbf{z}_{nt}^*) \in \mathbb{R}_+$ is the mixture PSDs with a sample $\mathbf{z}_{nt}^* \sim q_\phi(\mathbf{z}_{nt} \mid \mathbf{X})$. Because all the operations for calculating the ELBO are differentiable, the networks are optimized by using stochastic gradient ascent.

D. Source separation

Once the generative and inference models are trained, they are used to separate unseen mixture signals. Specifically, the inference model first estimates the model parameters $\mathbf{Q}$, $\mathbf{W}$, and $\hat{\mathbf{z}}_{ntd} = \boldsymbol{\mu}_{\phi,nt}(\mathbf{X})$. The source signal $\hat{s}_{nft}$ is then estimated by a multichannel Wiener filter as follows:

$$\hat{s}_{nft} \leftarrow \mathbf{u}^{\mathsf{T}}\mathbf{Y}_{nft}\mathbf{Y}_{:ft}^{-1}\mathbf{x}_{ft}, \quad (16)$$

TABLE I: Separation performance in SDR, PESQ, and STOI and elapsed time for separation in seconds.

Method	# of iters.	Elapsed time	Average			$K=2$			$K=3$			$K=4$		
			SDR	PESQ	STOI	SDR	PESQ	STOI	SDR	PESQ	STOI	SDR	PESQ	STOI
MNMF	200	2.07	7.5	1.49	0.76	13.0	1.93	0.85	8.3	1.47	0.79	3.9	1.26	0.69
ILRMA	200	1.36	7.0	1.43	0.76	13.2	1.83	0.86	7.7	1.39	0.79	3.2	1.24	0.69
FastMNMF	200	1.81	9.3	1.60	0.80	15.3	2.12	0.89	10.1	1.59	0.83	5.3	1.32	0.74
Neural FCA (fix z)	200	2.67	8.9	1.71	0.79	15.2	2.28	0.89	10.1	1.75	0.83	4.5	1.36	0.71
Neural FCA	5	0.14	8.0	1.48	0.78	14.0	1.90	0.88	9.1	1.47	0.82	3.8	1.26	0.69
Neural FCA	10	0.26	8.6	1.53	0.79	14.6	1.98	0.89	9.7	1.52	0.83	4.3	1.28	0.70
Neural FCA	100	2.40	10.6	1.81	0.83	16.2	2.35	0.90	11.8	1.87	0.86	6.5	1.46	0.76
Neural FCA	200	4.77	11.1	**1.88**	0.84	16.4	**2.41**	0.90	12.2	**1.95**	0.87	7.2	**1.52**	0.78
Neural FastFCA (ours)	–	**0.09**	**11.6**	1.85	**0.85**	**17.4**	**2.41**	**0.91**	**12.7**	1.90	**0.88**	**7.5**	1.50	**0.79**

where $\mathbf{u}$ is a one-hot vector representing a reference channel (the first channel in this paper), and $\mathbf{Y}_{:ft} = \sum_{n=1}^{N} \mathbf{Y}_{nft}$ is the sum of source images $\mathbf{Y}_{nft} = g_{\theta,f}(\hat{\mathbf{z}}_{ntd})\mathbf{Q}_f^{-1}\mathrm{diag}(\mathbf{w}_n)\mathbf{Q}_f^{-\mathsf{H}}$.

IV. Experimental Evaluation

The proposed method was evaluated with simulated mixture signals of various numbers of speech sources.

A. Dataset

We generated mixture signals of speech source signals by following the spatialized WSJ0-mix dataset [25]. Each mixture signal consisted of speech signals randomly selected from the WSJ0 English speech corpus [26]. We used the same subsets of speakers and utterances as in the WSJ0-mix dataset. In contrast to the WSJ0-mix dataset, the number of source signals was randomly selected from $K \in \{2, 3, 4\}$. A 6-channel microphone array ($M = 6$) with random configuration is located randomly around the center of a room having random dimensions between $5\,\mathrm{m} \times 5\,\mathrm{m} \times 3\,\mathrm{m}$ and $10\,\mathrm{m} \times 10\,\mathrm{m} \times 5\,\mathrm{m}$. The sound sources were also randomly located while keeping the distance between each other more than 1 m. The reverberation time (RT_{60}) was randomly sampled between 200 ms and 600 ms, and the room impulse response for each source was simulated with the image method. The speech signals were mixed at random powers uniformly chosen between -2.5 dB and $+2.5$ dB. White diffuse noise with a signal-to-noise ratio of 30 dB was added to each mixture signal as background noise. We generated 20,000, 5,000, and 3,000 mixture signals for training, validation, and test sets, respectively. All the mixtures were dereverberated by the weighted prediction error (WPE) method [27].

B. Experimental condition

The network architectures of the inference and generative models were experimentally determined as follows. The inference model consisted of ISS and DNN blocks with $B = 8$. Each DNN block consisted of a U-Net-like architecture [7], [28] having five 256-channel 1D-convolutional layers. Each layer had a kernel size of 5 and parametric rectified linear units (PReLUs). The input feature of the 0-th DNN block was the log-power spectrum and the inter-channel phase differences of an input mixture $\mathbf{x}_{ft}$. That of the b-th ($b \geq 1$) DNN block was a concatenation of the internal feature $\mathbf{h}_{\phi,t}^{(b-1)}$ and a 512-dimensional vector converted from the log-power spectrum of $\tilde{\mathbf{x}}_{ft}^{(b-1)}$ with a 1×1-convolution layer. We obtained the TF masks $m_{\phi,ftm}^{(b-1)}$ and $\boldsymbol{\omega}_{\phi,nft}$ with a sigmoid function and $\sigma_{\phi,ntd}^2$ with a softplus function. The generative model $g_{\theta,f}$, on the other hand, consisted of three layers of 256-channel 1×1-convolutional layers with PReLUs as in [18].

We trained the inference and generative models with an Adam optimizer [23] for 200 epochs with a learning rate of 1.0×10^{-3}. The spectrograms were obtained using the short-time Fourier transform with a window length of 512 samples and a hop length of 128 samples. The training was performed by splitting the spectrograms into 500-frame clips, and the batch size was set to 128 clips. The dimension of the latent features was set to $D = 50$. The number of sound sources was set to $N = 5$, assuming the maximum number of sources and diffuse noise ($4 + 1$). We performed the cyclic annealing of the KL term in the ELBO [18], [29]. The diagonalizer was initialized to an identity matrix $\mathbf{Q}_f^{(0)} \leftarrow \mathbf{I}$. These hyperparameters were empirically determined using the validation set.

Our method was compared with existing BSS methods and the original neural FCA. As BSS methods, we evaluated MNMF [10], ILRMA [21], and FastMNMF [11]. The number of sources for MNMF and FastMNMF was set to 5. The numbers of bases and iterations for all the methods were set to 16 and 200, respectively. The SCMs for MNMF were initialized by ILRMA. The neural FCA [18] had the same generative model $g_{\theta,f}$ as our method, and its inference model consisted of nine U-Net-like blocks to have the same number of blocks as our method. We trained the neural FCA with the same hyperparameters as the proposed method. The EM inference for SCMs $\mathbf{H}_{nf}$ was iterated 5 times in the training phase following the literature [18]. At the test phase, the SCMs $\mathbf{H}_{nf}$ and latent features $\mathbf{z}_{nt}$ were updated to fit the observation with the EM rule and an Adam optimizer, respectively. The learning rate for Adam was set to 0.2. We evaluated different numbers of iterations (5, 10, 100, and 200 times) to assess how many iterations were needed for the conversion.

The performance was evaluated in terms of the signal-to-distortion ratio (SDR) [30] in dB, the perceptual evaluation of speech quality (PESQ) [31], and the short-term objective intelligibility (STOI) [32]. They were evaluated with K separated signals having the highest powers in the separation results. We also measured the elapsed time for separating a 5-second clip on an NVIDIA V100 accelerator with Intel Xeon Gold 6148

Processor. To fully utilize the accelerator, the conventional BSS methods were implemented with CuPy 11.5.0, and the neural methods were implemented with PyTorch 1.13.1.

C. Experimental results

The separation performance for each number of sources $K \in \{2, 3, 4\}$ was summarized in TABLE I. We first see that the original neural FCA required 200 times of updates for convergence. Besides, the neural FCA deteriorated by fixing the latent features $\mathbf{z}_{nt}$ to the output of the inference model ("fix z" in the table). This result indicates that the inference model failed to precisely estimate the latent source features only from the observed mixture. In contrast, our neural FastFCA, which does not update the outputs of the inference model, outperformed the neural FCA in SDR and STOI and outperformed that without updates of $\mathbf{z}_{nt}$ in all the metrics. In addition, our method reduced the computational time to about 2 % of that for the neural FCA (4.77 [s]). The proposed method also clearly outperformed the conventional BSS methods of MNMF, ILRMA, and FastMNMF for all the conditions of $K \in \{2, 3, 4\}$. We would also note that the neural FastFCA was trained successfully by using multichannel mixture signals to have different numbers of sources. This result shows the promising possibility of our method to train a neural separation model in an unsupervised manner by specifying the maximum number of sources in the training mixtures.

V. Conclusion

This paper presented a neural BSS method called neural FastFCA based on the integration of the neural source model, JD spatial model, and ISS-based inference model. Specifically, we extended the original neural FCA to have a JD full-rank spatial model to efficiently reduce the computational cost. Our neural FastFCA also introduces an ISS-based inference model to improve the separation performance. The experimental results with mixture signals having two to four sources showed that our neural FastFCA outperformed existing BSS methods. In addition, the elapsed time for performing our method was reduced to 2 % of that for the original neural FCA. Our future work includes further extending our method with various BSS techniques. For example, the joint dereverberation and separation of moving sources is an important feature to computationally understand mixture signals recorded in indoor environments. We will also investigate separating various kinds of sound sources in addition to speech signals.

References

[1] S. Watanabe *et al.*, "CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," *arXiv preprint arXiv:2004.09249*, 2020.

[2] J. Du *et al.*, "The USTC-NELSLIP systems for CHiME-6 challenge," in *Proc. CHiME-6 Workshop*, 2020, pp. 1–5.

[3] A. S. Subramanian *et al.*, "Far-field location guided target speech extraction using end-to-end speech recognition objectives," in *Proc. IEEE ICASSP*, 2020, pp. 7299–7303.

[4] R. Scheibler *et al.*, "Sound event localization and detection with pretrained audio spectrogram transformer and multichannel separation network," in *DCASE Workshop*, 2022, pp. 1–5.

[5] N. Turpault *et al.*, "Improving sound event detection in domestic environments using sound separation," in *Detection and Classification of Acoustic Scenes and Events (DCASE) Workshop*, 2020, pp. 1–5.

[6] J. Zhu *et al.*, "Multi-decoder DPRNN: Source separation for variable number of speakers," in *Proc. ICASSP*, 2021, pp. 3420–3424.

[7] E. Tzinis *et al.*, "Sudo rm-rf: Efficient networks for universal audio source separation," in *Proc. IEEE MLSP*, 2020, pp. 1–6.

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

[9] N. Ono, "Stable and fast update rules for independent vector analysis based on auxiliary function technique," in *Proc. IEEE WASPAA*, 2011, pp. 189–192.

[10] H. Sawada *et al.*, "Multichannel extensions of non-negative matrix factorization with complex-valued data," *IEEE TASLP*, vol. 21, no. 5, pp. 971–982, 2013.

[11] K. Sekiguchi *et al.*, "Fast multichannel nonnegative matrix factorization with directivity-aware jointly-diagonalizable spatial covariance matrices for blind source separation," *IEEE/ACM TASLP*, vol. 28, pp. 2610–2625, 2020.

[12] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE/ACM TASLP*, vol. 18, no. 3, pp. 550–563, 2009.

[13] N. Ito and T. Nakatani, "FastMNMF: Joint diagonalization based accelerated algorithms for multichannel nonnegative matrix factorization," in *Proc. IEEE ICASSP*, 2019, pp. 371–375.

[14] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[15] K. Sekiguchi *et al.*, "Semi-supervised multichannel speech enhancement with a deep speech prior," *IEEE/ACM TASLP*, vol. 27, no. 12, pp. 2197–2212, 2019.

[16] S. Leglaive *et al.*, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *Proc. IEEE ICASSP*, 2019, pp. 101–105.

[17] H. Kameoka *et al.*, "Semi-blind source separation with multichannel variational autoencoder," *arXiv preprint arXiv:1808.00892*, 2018.

[18] Y. Bando *et al.*, "Neural full-rank spatial covariance analysis for blind source separation," *IEEE SPL*, vol. 28, pp. 1670–1674, 2021.

[19] R. Scheibler and M. Togami, "Surrogate source model learning for determined source separation," in *Proc. IEEE ICASSP*, May 2021, pp. 176–180.

[20] K. Saijo and R. Scheibler, "Spatial loss for unsupervised multi-channel source separation," in *Proc. Interspeech*, Sep. 2022, pp. 241–245.

[21] D. Kitamura *et al.*, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," *IEEE/ACM TASLP*, vol. 24, no. 9, pp. 1626–1641, 2016.

[22] R. Scheibler and N. Ono, "Fast and stable blind source separation with rank-1 updates," in *Proc. IEEE ICASSP*, 2020, pp. 236–240.

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

[24] N. Q. K. Duong *et al.*, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *IEEE/ACM TASLP*, vol. 18, no. 7, pp. 1830–1840, 2010.

[25] Z.-Q. Wang *et al.*, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *Proc. IEEE ICASSP*, 2018, pp. 1–5.

[26] J. Garofolo *et al.*, "CSR-I (WSJ0) Complete LDC93S6A," DVD, 2007, Philadelphia: Linguistic Data Consortium.

[27] T. Yoshioka and T. Nakatani, "Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening," *IEEE TASLP*, vol. 20, no. 10, pp. 2707–2720, 2012.

[28] Y. Bando *et al.*, "Weakly-Supervised Neural Full-Rank Spatial Covariance Analysis for a Front-End System of Distant Speech Recognition," in *Proc. Interspeech 2022*, 2022, pp. 3824–3828.

[29] H. Fu *et al.*, "Cyclical annealing schedule: A simple approach to mitigating KL vanishing," in *Proc. NAACL-HLT*, 2019, pp. 240–250.

[30] E. Vincent *et al.*, "Performance measurement in blind audio source separation," *IEEE TASLP*, vol. 14, no. 4, pp. 1462–1469, 2006.

[31] A. W. Rix *et al.*, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE ICASSP*, vol. 2, 2001, pp. 749–752.

[32] C. H. Taal *et al.*, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *Proc. IEEE ICASSP*, 2010, pp. 4214–4217.