FREQUENCY DOMAIN BLIND SOURCE SEPARATION EXPLOITING HIGHER-ORDER DEPENDENCIES

Taesu Kim[1,2]*, Hagai Attias*[3]*, Soo-Young. Lee*[2]*, and Te-Won Lee*[1]

[1]Institute for Neural Computation, University of California, San Diego, La Jolla, CA 92093, USA
[2]Deptartment of Biosystems, Korea Advanced Institute of Science and Technology, Daejeon 305-701, KOREA
[3]Golden Metallic Inc, P.O. Box 475608, San Francisco, CA 94147, USA

ABSTRACT

We propose a novel approach to the blind source separation (BSS) that exploits frequency dependencies within a source. In contrast to conventional algorithms that separate the sources independently in each frequency bin, we assume that dependencies exist between frequency bins in a source signal. In this manner, we can reduce or eliminate the well-known frequency permutation problem. We derive the learning algorithm by defining a cost function as an extension of mutual information between multivariate random variables and by introducing a source prior that models the inherent frequency dependencies. This results in a simple form of a multivariate score function. In simulations and real recording experiments, we evaluate the performance of the proposed method and compare it against other well-known algorithms under various conditions. Our results indicate that modeling dependencies yields improved performance and robust scaling to higher number of sources and mixtures.

1. INTRODUCTION

For the separation of convolved and time delayed sources from mixtures of recordings, researchers have proposed several time and frequency domain based algorithms. The benefit of frequency domain algorithms is that it can handle longer filter lengths with reasonable computational complexity since the multiplication at each frequency bin replaces convolution operation in the time domain. Thus, one can apply a standard ICA algorithm to instantaneous mixtures in each frequency bin. A well known problem that occurs in this approach is the frequency source permutation that needs to be solved correctly to reconstruct the desired source signal in time. Various approaches have been proposed to solve the permutation problem. A popular approach is to impose a smoothness constraint of the source that translates into smoothing the separating filter. This approach has been realized by several techniques such as averaging separating matrices with adjacent frequencies [1], limiting the filter length in the time domain [2], or considering the coherency of separating matrices at adjacent frequencies [3]. Another related approach is based on direction of arrival (DOA) estimation which is much used in array signal processing. By analyzing the directivity patterns formed by a separating matrix, source directions can be estimated and therefore permutations can be aligned [4]. When the sources are nonstationary signals, one can employ the inter-frequency correlations of signal envelopes to align permutations [5]. Although these methods perform well under certain specific conditions, there is no method that performs well in general conditions. Moreover, in the case of an ill-posed problem, e.g. the case that each mixing filter of the source is very similar, the sources are located close to each other, or DOA of the sources are similar, the methods developed so far fail to separate the source signals.

In this paper, we propose a novel approach for BSS by focusing on a new cost function, a dependency model, and a multivariate score function, which captures inter-frequency dependencies in data. These dependencies are related to an improved model for the source signal prior. While the source priors are defined as independent priors at each frequency bin in conventional algorithms, we utilize higher-order dependencies across frequency. In this manner, we define each source model as a Gaussian scale mixture, which models variance dependencies. The algorithm itself is able to preserve variance dependencies and structures of frequencies. Therefore, the permutation problem is avoided, and the separation performances are comparably high even in severely ill-posed conditions.

2. PROPOSED METHOD

The proposed method consists of the mixing and separating procedure in a convolutive environment, the definition of a cost function, and an algorithm for learning the parameters of the separating filters. We define a new cost function for the frequency domain BSS, and derive its learning algorithm by minimizing it. Notation used in this paper is defined below[1].

2.1. Model

We define the relationship between the sources and observations. Let $x_i(t)$ be the ith observation signal at time t.

$$x_i(t) = \sum_{j=1}^{L} \sum_{\tau=0}^{T-1} h_{ij}(\tau) s_j(t-\tau), \tag{1}$$

T. Kim and S.-Y. Lee were supported by the CHUNG Moon Soul Center for BioInformation and BioElectronics and also by the Brain Neuroinformatics Research Program from Korean Ministry of Science and Technology.

[1]We use plain lower-cased characters to denote scalar variables, boldfaced, lower-cased characters to denote vector variables, and upper-cased characters to denote matrix variables. Super-script indicates a frequency bin, and sub-script indicates a source or an observation. For example, $\mathbf{x}_i$ is the ith observation vector that consists of K frequency bins, $\left[x_i^{(1)}, \cdots, x_i^{(K)}\right]^{\mathsf{T}}$. $\mathbf{x}^{(k)}$ is an observation vector at the kth frequency bin, which consists of M observations at the kth frequency bin, $\left[x_1^{(k)}, \cdots, x_M^{(k)}\right]^{\mathsf{T}}$. $H^{(k)} \equiv \left\{h_{ij}^{(k)}\right\}$ means that $h_{ij}^{(k)}$ is the ith row, jth column element of the matrix $H^{(k)}$. $x_i^{(k)}[n]$ denotes the nth sample of random variable $x_i^{(k)}$. $x_i^{\star(k)}$ denotes the complex conjugate of $x_i^{(k)}$, and $\mathbf{x}_i^{\dagger}$ denotes the conjugate transpose of $\mathbf{x}_i$.

where $h_{ij}(t)$ is a time domain transfer function from the jth source to the ith observation, which has T length in time, $s_j(t)$ is the jth source signal at time t, and L is the number of sources. If the window length, K, is sufficiently longer than the length of the mixing filter $h_{ij}(t)$, the convolution in the time domain is approximately converted to multiplication in the frequency domain as follows.

$$x_i^{(k)}[n] \approx \sum_{j=1}^{L} h_{ij}^{(k)} s_j^{(k)}[n] \tag{2}$$

If the separating filter matrices exist, that is, the inverses or pseudo-inverses of mixing matrices at each frequency exist ($L \leq M$), then the separated ith source signal is obtained by

$$\hat{s}_i^{(k)}[n] = \sum_{j=1}^{M} g_{ij}^{(k)} x_j^{(k)}[n] \approx s_i^{(k)}[n], \tag{3}$$

where $g_{ij}^{(k)}$ is the separating filter coefficient at the kth frequency bin, and M is the number of observed signals.

2.2. Cost function

Instead of separating the sources independently at each frequency, our algorithm separates the sources as vector signals. In order to separate multivariate sources from multivariate observations, we need to define a cost function for multivariate random variables. Here, we define Kullback-Leibler divergence between two functions as the measure of independence. One is an exact joint probability density function, $p\left(\hat{\mathbf{s}}_1, \cdots, \hat{\mathbf{s}}_L\right)$, and the other is a nonlinear function which is the product of approximated probability density functions of individual source vectors, $\prod_{i=1}^{L} q\left(\hat{\mathbf{s}}_i\right)$. We would think of it as an extension of mutual information between multivariate random variables.

$$\begin{aligned} \mathcal{C} &= \mathcal{KL}\left(p\left(\hat{\mathbf{s}}_1, \cdots, \hat{\mathbf{s}}_L\right) \parallel \prod_{i=1}^{L} q\left(\hat{\mathbf{s}}_i\right)\right) \\ &= -\sum_{k=1}^{K} \log|\det G^{(k)}| - \sum_{i=1}^{L} E \log q(\hat{\mathbf{s}}_i) + const. \end{aligned} \tag{4}$$

Note that the random variables in above equations are multivariate. The interesting parts of this cost function are that each source is multivariate and it would be minimized when dependencies between the source vectors is removed, but dependencies between the elements of each vector does not need to be removed. Therefore, the cost function preserves the inherent frequency dependency within each source, but it removes dependency between the sources.

2.3. Learning Algorithm: a gradient descent method

Now that we defined the cost function, derivation of the learning algorithm is straightforward. Here, we are using a gradient descent method to minimize the cost function. By differentiating the cost function $\mathcal{C}$ with respect to each coefficient of the separating matrices, $g_{ij}^{(k)}$, we can obtain the gradients for the coefficients as follows.

$$\Delta g_{ij}^{(k)} = -\frac{\partial \mathcal{C}}{\partial g_{ij}^{(k)}} = g_{ij}^{-\dagger(k)} - E\varphi^{(k)}\left(\hat{s}_i^{(1)}, \cdots, \hat{s}_i^{(K)}\right) x_j^{\star(k)} \tag{5}$$

where $\left(G^{(k)^{-1}}\right)^{\dagger} \equiv \left\{g_{ij}^{-\dagger(k)}\right\}$. By multiplying scaling matrices, $G^{(k)^{\dagger}} G^{(k)}$, to the gradient matrices, $\Delta G^{(k)} \equiv \left\{\Delta g_{ij}^{(k)}\right\}$, we can obtain the natural gradient, which is well known as a fast convergence method [6]

$$\Delta g_{ij}^{(k)} = \sum_{l=1}^{L} \left(I_{il} - E\varphi^{(k)}\left(\hat{s}_i^{(1)}, \cdots, \hat{s}_i^{(K)}\right) \hat{s}_l^{\star(k)}\right) g_{lj}^{(k)} \tag{6}$$

where I_{il} is 1 only when $i = l$, otherwise 0, and the nonlinear function, $\varphi^{(k)}(\cdot)$, is given as

$$\varphi^{(k)}\left(\hat{s}_i^{(1)}, \cdots, \hat{s}_i^{(K)}\right) = -\frac{\partial \log q\left(\hat{s}_i^{(1)}, \cdots, \hat{s}_i^{(K)}\right)}{\partial \hat{s}_i^{(k)}} \tag{7}$$

We would term it a multivariate score function corresponding to a score function in the conventional ICA. From above derivation, one can notice that the only difference between our approach and the conventional ICA algorithm is the form of a score function. If we define the multivariate score function as a single-variate function such as $\varphi\left(\hat{s}_i^{(k)}\right) = \exp\left(j \cdot \arg\left(\hat{s}_i^{(k)}\right)\right)$, the algorithm is converted to the same as the conventional ICA. Therefore, the key point of our algorithm is to define a proper multivariate score function, which can capture higher-order frequency dependencies. Since the cost function includes $q\left(\hat{\mathbf{s}}_i\right)$, which is an approximated source prior, that is, $q\left(\mathbf{s}_i\right) \approx p\left(\mathbf{s}_i\right)$, a multivariate score function is closely related to a source prior, and it can be obtained by differentiating log prior with respect to the each element of a source vector. It is clear that a single-variate score function results from independent prior. However, instead of defining independency, we assume the source prior as a higher-orderly dependent distribution, which can be generally written as follows

$$p\left(\mathbf{s}_i\right) = \alpha \cdot f\left(\delta_\lambda\left(\mathbf{s}_i\right)\right), \tag{8}$$

where α is a normalization term, $\delta_\lambda\left(\mathbf{s}_i\right) = \left(\sum_k \left|s_i^{(k)}\right|^\lambda\right)^{1/\lambda}$, and $f(\cdot)$ is an arbitrary function. Thus, the multivariate score function is given as follows.

$$\varphi^{(k)}\left(\hat{s}_i^{(1)}, \cdots, \hat{s}_i^{(K)}\right) = -\frac{f'\left(\delta_\lambda\left(\mathbf{s}_i\right)\right)}{f\left(\delta_\lambda\left(\mathbf{s}_i\right)\right)} \cdot \delta'_\lambda\left(\mathbf{s}_i\right) \tag{9}$$

For example, to obtain a dependent multivariate super-Gaussian distribution, we may choose $\lambda = 2$ and $f(\cdot) = \exp(\cdot)$, which result in a simple multivariate score function as follows.

$$\varphi^{(k)}\left(\hat{s}_i^{(1)}, \cdots, \hat{s}_i^{(K)}\right) = \frac{\hat{s}_i^{(k)}}{\delta_2\left(\mathbf{s}_i\right)} \tag{10}$$

Since the form of a multivariate score function is related to dependency of sources, the proper form of a multivariate score function might vary with different types of dependency, as apparent to one having ordinary skill in the art.

Although our approach avoids the permutation problem by exploiting higher-order frequency dependencies, the scaling problem still need to be solved. A well-known method is obtained by the minimal distortion principle [7], in which we can obtain the separating filter matrix that has reasonable scales by replacing it as follows.

$$G^{(k)} \leftarrow diag\left(G^{-1}(k)\right) G^{(k)}, \tag{11}$$

where $diag\left(X\right)$ denotes the diagonal matrix of the matrix X. After solving the scaling problem, we calculate the finally separated sources in the frequency domain by (3). Then, we perform an inverse Fourier transform and overlap add to reconstruct time domain signals.

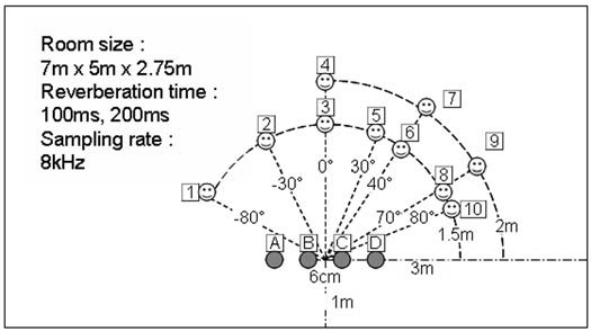


Fig. 1. Simulated room environments. All the heights of sources and microphones were $1.5m$. Several combinations of source locations were selected. 2 microphones (B,C) were used for 2 sources case and 4 microphones (A,B,C,D) were used for 4 sources case.

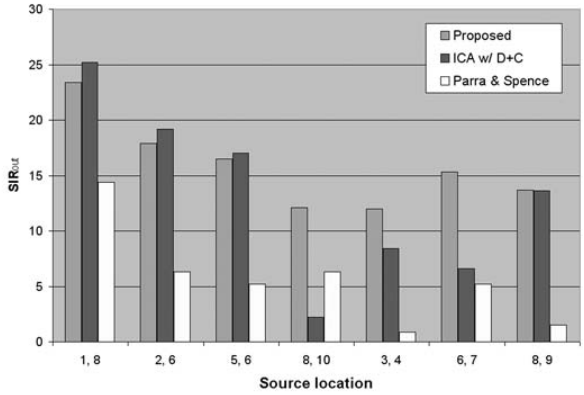
3. EXPERIMENTAL RESULTS

We evaluated the performance of the proposed algorithm using both simulated and real data. Simulated data were obtained by simulating impulse responses of a rectangular room based on the image model technique. To generate the microphone signals, we used 8 second-long audio signals sampled at 8kHz, and they were convolved with corresponding room impulse responses. The proposed algorithm was compared with 2 well-known frequency domain BSS algorithms such as ICA algorithm with permutation correction, and Parra and Spence's algorithm[2] [2]. The former method was obtained by using the conventional score function. To solve the permutation problem in this method, we used the method to consider DOA and inter-frequency correlations together [8]. Parra and Spence's algorithm avoids the permutation problem by limiting the length of the filter in the time domain to smooth the shape of the filter in the frequency domain, while learning the separating filters. The performances were measured by signal to interference ratio (SIR) in dB. Real data were obtained in an ordinary conference room, where human speakers read several sentences and loud speaker played music. In all experiments, we used a 1024 point FFT and Hanning window to convert time domain signals to the frequency domain. The length of window was 1024 samples and shift size was 256 samples. Initial values for both the proposed and the conventional ICA based algorithm were chosen as whitening matrix in each frequency bin. The algorithm ran until the decrement of the cost function was less than 10^{-6}. For Parra and Spence's algorithm, we used the same number of FFT points and limited the length of time domain filter to 256, which provided the best performances.

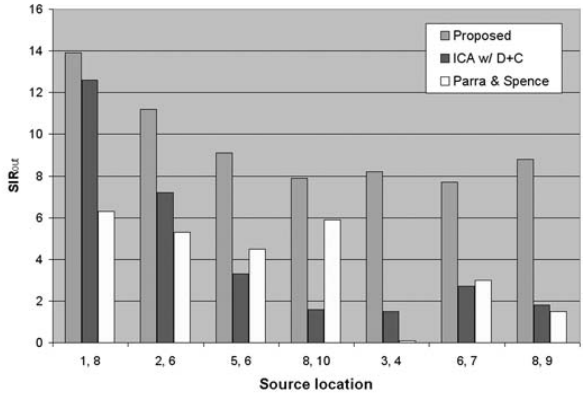
First, the proposed algorithm was applied to the problem with 2 microphones and 2 sources in simulated room environments. We assumed that the room size was 7m×5m×2.75m. For an intensive analysis, we evaluated the performances with a number of source locations and reverberation times. All the heights of sources and microphones were 1.5m. The environments are shown in Fig. 1, in which microphone B and C were used, and we chose 7 pairs of source locations. Although 2 cases of locations such as 1 and 8, and 2 and 6 are comparably easier cases, 5 and 6, and 8 and 10 are more difficult cases because the sources are located on the same side and have similar DOAs. The other 3 cases such as 3 and 4, 6 and 7, and 8 and 9 are the most difficult cases, because the sources are located closely as well as they have the same DOAs. Fig. 2 shows the results of all cases with varying reverberation times, when one source was a male speech, and the other was a female speech. The average $\mathrm{SIR}_{\mathrm{in}}$ of all cases was about 0dB. To evaluate the computational complexity, we measured the computational time when the sources were located 1 and 8 and the reverberation time was 100ms. The proposed algorithm was coded in MATLAB and executed on Intel Pentium 4 3GHz processor. TABLE 1 shows the cpu time per iteration and the number of iteration until convergence. The computational time per iteration is almost similar to the ICA algorithm. Since the learning rule in each frequency bin is related to others, the convergence speed is a little slower.

Secondly, we tested the algorithms with 4 microphones and 4 sources. The environments were the same as the previous experiments, except for the locations of the sources and microphones. In this experiment, all the microphones in Fig. 1 were used, and 5 combinations of source locations were chosen. The sources were 2 male

[2]The code was downloaded from *http://ida.first.gmd.de/∼harmeli /download/download_convbss.html*



(a) 100ms reverberation

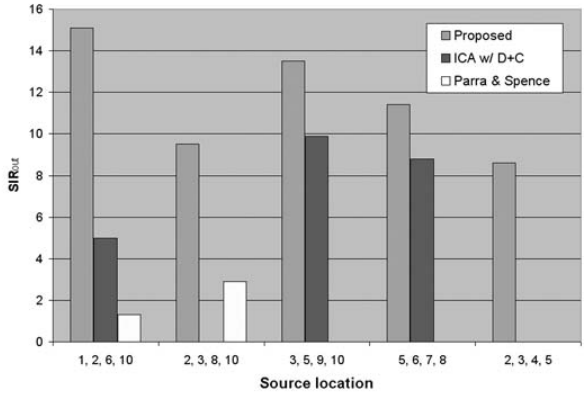


(b) 200ms reverberation

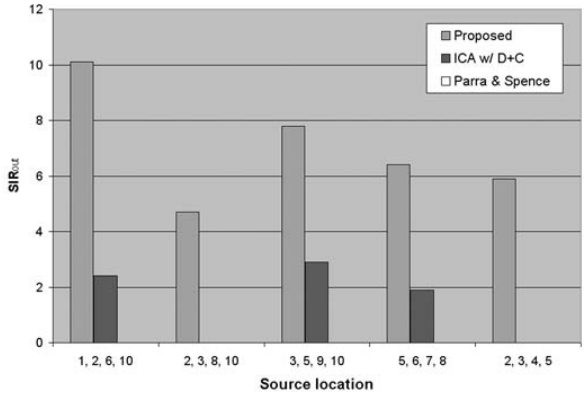
Fig. 2. Experimental results of 2 sources separation. $\mathrm{SIR}_{\mathrm{out}}$ was compared in various pairs of source locations and reverberation time with 2 sources and 2 microhpones (B,C), which are shown in Fig. 1

Table 1. Computational Time

	Time per iter.	# of iter.	Learning time
Proposed	0.1689s	200	33.78s
ICA	0.1623s	130	21.1s



(a) 100ms reverberation



(b) 200ms reverberation

Fig. 3. Experimental results of 4 sources separation. $\mathrm{SIR}_{\mathrm{out}}$ was compared in various pairs of source locations and reverberation time with 4 sources and 4 microphones (A,B,C,D), which are shown in Fig. 1. In some cases, the other 2 algorithms failed to separate sources, where the bar was not shown

speeches and 2 female speeches. Fig. 3 shows the results of all cases, where the average $\mathrm{SIR}_{\mathrm{in}}$ of all cases was about -5dB. As shown in Fig. 2 and 3, the proposed algorithm outperforms the others in most cases. Even in the worst cases, the others did not exceed the proposed algorithm more than 2dB. The ICA algorithm with permutation correction based on inter-frequency correlation is not robust, because a misalignment of a permutation at a certain frequency bin may cause consecutive misalignments of neighboring frequency bins. By combining DOA and correlation, robustness can be improved a little. But the DOA based permutation correction is not precise, as the reverberation time increases or the source locations are close. Moreover, it completely fails when the sources have the same DOAs. So, the algorithm combining DOA and correlation is still not robust enough in some cases. Its performance can be severely bad in some cases although it performs best in certain cases. The severe disadvantage of Parra and Spence's algorithm is that it cannot use the full length of the filter, because it limits the filter length to avoid the permutation problem. Thus, the effective filter length in the time domain was 256, even though we used a 1024 point FFT. The performances of their algorithm were degraded more than others, as the source locations were difficult or the reverberation time was long. However, the proposed algorithm overcomes these disadvantages. Therefore, it does not limit the filter length, and it is very robust.

Finally, we recorded real data in an ordinary conference room that has long reverberation time. 4 microphones were located in a line. The sources consisted of 3 human speakers reading sentences, and a hip-hop music from a loud speaker, which were located approximately 1m$\sim$2m from the microphones. 3 human speakers were located approximately 1m$\sim$2m from the microphones, and read several sentences. The approximate SIR improvement was about 14dB. Audio files and more information are available on our web page.[3]

4. CONCLUSIONS

We have proposed a new algorithm for BSS that utilizes not only independencies between the sources but also frequency dependencies within source signals. Instead of defining independence for each frequency bin, we have assumed that a source signal in the frequency domain has frequency dependencies, which resulted in a multivariate score function. This approach solves the permutation problem and yields very low complexity. Experimental results have shown that the proposed algorithm is robust and precise in many challenging cases. We believe that exploiting higher-order frequency dependencies within the source signal is a key to solving the BSS problem in a real environment.

5. REFERENCES

[1] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21–34, 1998.

[2] L. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 3, pp. 320–327, 2000.

[3] F. Asano, S. Ikeda, M. Ogawa, H. Asoh, and N. Kitawaki, "A combined approach of array processing and independent component analysis for blind separation of acoustic signals," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2001, pp. 2729–2732.

[4] M. Z. Ikram and D. R. Morgan, "A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2002, pp. 881–884.

[5] N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocomputing*, vol. 41, pp. 1–24, 2001.

[6] S.-I. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Adv. Neural information Processing Systems*, 1996, vol. 8.

[7] K. Matsuoka and S. Nakashima, "Minimal distortion principle for blind source separation," in *Proc. Int. Conf. on Independent Component Analysis and Blind Source Separation*, 2001, pp. 722–727.

[8] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech Audio Processing*, vol. 12, no. 5, pp. 530–538, 2004.

[3] *http://ergo.ucsd.edu/∼taesu/source_separation.html*