LETTER **Communicated by Aapo Hyvärinen**

Independent Vector Analysis for Source Separation Using a Mixture of Gaussians Prior

Jiucang Hao
jhao@ucsd.edu
Computational Neurobiology Laboratory, The Salk Institute for Biological Studies, La Jolla, CA, 92037, U.S.A.

Intae Lee
intelli@ucsd.edu
Institute for Neural Computation, University of California at San Diego, CA, 92093, U.S.A.

Te-Won Lee
tewon@qualcomm.com
Qualcomm, San Diego, CA, 92121, U.S.A.

Terrence J. Sejnowski
terry@salk.edu
Howard Hughes Medical Institute at the Salk Institute, La Jolla, CA, 92037, and Division of Biological Sciences, University of California at San Diego, CA, 92093, U.S.A.

Convolutive mixtures of signals, which are common in acoustic environments, can be difficult to separate into their component sources. Here we present a uniform probabilistic framework to separate convolutive mixtures of acoustic signals using independent vector analysis (IVA), which is based on a joint distribution for the frequency components originating from the same source and is capable of preventing permutation disorder. Different gaussian mixture models (GMM) served as source priors, in contrast to the original IVA model, where all sources were modeled by identical multivariate Laplacian distributions. This flexible source prior enabled the IVA model to separate different type of signals. Three classes of models were derived and tested: noiseless IVA, online IVA, and noisy IVA. In the IVA model without sensor noise, the unmixing matrices were efficiently estimated by the expectation maximization (EM) algorithm. An online EM algorithm was derived for the online IVA algorithm to track the movement of the sources and separate them under nonstationary conditions. The noisy IVA model included the sensor noise and combined denoising with separation. An EM algorithm was developed that found the model parameters and separated the sources simultaneously.

These algorithms were applied to separate mixtures of speech and music. Performance as measured by the signal-to-interference ratio (SIR) was substantial for all three models.

1 Introduction

Blind source separation (BSS) addresses the problem of recovering original sources from mixtures, knowing only that the mixing processes is linear. The applications of BSS include speech separation, cross-talk elimination in telecommunications, and electroencephalograph (EEG) and magneto-encephalograph (MEG) data analysis.

Independent component analysis (ICA) (Comon, 1994; Lee, 1998; Hyvärinen, Karhunen, & Oja, 2001) is effective in separating sources when the mixing process is linear and the sources are statistically independent. One natural way to characterize the independence is by using a factorized source prior, which requires knowing the probability density function (PDF) for sources. The Infomax algorithm (Bell & Sejnowski, 1995) used a supergaussian source prior that was effective for many natural sources. The extended Infomax (Lee, Girolami, & Sejnowski, 1999) could also separate sources with subgaussian statistics. A gaussian mixture model (GMM), introduced as flexible source priors in Moulines, Cardoso, and Gassiat (1997), Attias (1999), Attias, Deng, Acero, and Platt (2001), and Attias, Platt, Acero, and Deng (2000) can be directly estimated from the mixtures. A nonparametric density estimator has also been used for ICA (Boscolo, Pan, & Roychowdhury, 2004) and higher-order statistics are an alternative characterization of independence and are distribution free (Cardoso, 1999; Hyvärinen & Oja, 1997; Hyvärinen, 1999). Other approaches have used kernels (Bach & Jordan, 2002), subspaces (Hyvärinen & Hoyer, 2000) and topographic neighborhoods (Hyvärinen, Hoyer, & Inki, 2001).

Speech separation is an example of mixing where the mixing process is a convolution (Lee, Bell, & Lambert, 1997; Mitianoudis & Davies, 2003; Torkkola, 1966). In some cases, the sources can be separated by ICA in the frequency domain, where the mixtures are approximately linear in every frequency bin. Because ICA is blind to the permutation, the separated frequency bins need to be aligned. This is called the permutation problem. One approach is to enforce the smoothness of the separated sources and the separation filters, for example, by comparing the separation matrices of neighbor frequencies (Smaragdis, 1998) and limiting the time-domain filter length (Parra & Spence, 2000). The permutation can also be aligned according to the direction of arrival (DOA), which can be estimated from the separation matrices (Kurita, Saruwatari, Kajita, Takeda, & Itakura, 2000). Cross-correlation of the frequency components has also been used to correct the permutations (Murata, Ikeda, & Ziehe, 2001).

A more direct approach to the permutation problem is to prevent the permutation from occurring instead of postprocessing to correct them. Independent vector analysis (IVA) (Kim, Attias, Lee, & Lee, 2007; Lee & Lee, 2007; Lee, Kim, & Lee, 2007), does this by exploiting the dependency among the frequency components. IVA assumed that the frequency components originating from the same source were dependent and that the frequency components originating from different sources were independent. The joint PDF of frequency components from each source was a multivariate distribution that captured the dependency across frequencies and prevented permutation disorder. By treating the frequency bins of each source as a vector, IVA captured the dependence within the vector, assuming that the different vectors were independent. IVA used a multivariate Laplacian distribution as source priors, and the unmixing matrices were estimated using maximum likelihood by gradient ascent algorithm. Due to the dependency modeling, the separation for all frequency bins was done collectively. However, the statistical properties of the sources could be different, and the Laplacian PDF may not be accurate for all the sources. Also IVA assumed no sensor noise, which is not realistic in real environments.

In this letter, we propose a general probabilistic framework for IVA to separate convolved acoustic signals. The frequencies from the same source were jointly modeled by a GMM, which captured the dependency and prevented permutation. Different sources were modeled by different GMMs, which enabled IVA to separate different type of sources. We considered three conditions: noiseless IVA, online IVA, and noisy IVA. Noiseless IVA assumed no sensor noise, similar to most ICA and IVA algorithms. Online IVA was capable of tracking moving sources and separating them, which is particularly useful in dynamic environments. Noisy IVA included the sensor noise and allowed speech denoising to be achieved together with source separation. Model parameters were estimated by maximum likelihood. Efficient expectation maximization (EM) algorithms were proposed for all conditions.

This letter is organized as follows. In section 2 we present the IVA model under a general probabilistic framework. Section 3 presents the EM algorithm for noiseless IVA. Section 4 presents an online EM algorithm for noiseless IVA. Section 5 presents the EM algorithm for noisy IVA. The experimental results are demonstrated in section 6 with simulations. Section 7 concludes the letter.

2 Independent Vector Analysis Model

2.1 Acoustic Model for Convolutive Mixing. We focus on the 2×2 problem: two sources and two microphones. Some of the algorithms can be generalized to multiple sources or microphones. Let $x_j[t]$ be the sources j and $y_l[t]$ be the channel l, at time t. The mixing process can be accurately described by the convolution. We consider both noisy case and noiseless

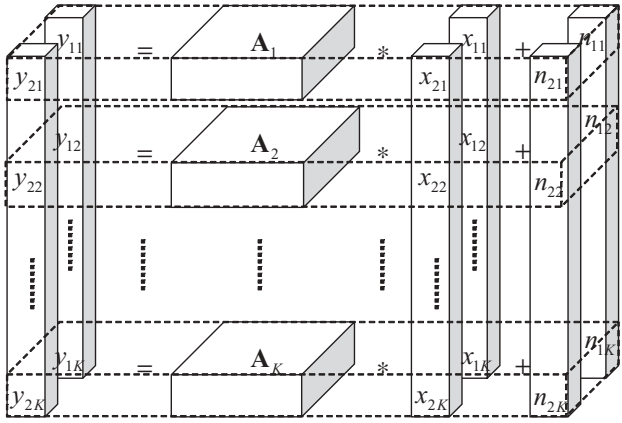

Figure 1: Mixture model of independent vector analysis (IVA). Dependent source components across the layers of linear mixtures are grouped into a multidimensional source, or vector.

case here,

$$\text{Noiseless IVA: } y_l[t] = \sum_{j=1}^{2} \sum_{\tau} h_{lj}[t](\tau) x_j[t-\tau] \tag{2.1}$$

$$\text{Noisy IVA: } y_l[t] = \sum_{j=1}^{2} \sum_{\tau} h_{lj}[t](\tau) x_j[t-\tau] + n_l[t], \tag{2.2}$$

where $h_{lj}[t]$ is time domain transfer function from the jth source to the lth channel, and $n_i[t]$ is the noise. Although the noiseless IVA is a special case of noisy IVA by setting $n_i[t] = 0$, the algorithms are quite different and treated separately.

Let k denote the frequency bin and $\mathbf{Y}_{kt} = (Y_{1kt}, Y_{2kt})^T$, $\mathbf{X}_{kt} = (X_{1kt}, X_{2kt})^T$, $\mathbf{N}_{kt} = (N_{1kt}, N_{2kt})^T$, be the vectors of the kth FFT coefficients of the mixed signals, the sources, and the sensor noise, respectively. When the fast Fourier transform (FFT) is applied, the convolution becomes multiplicative,

$$\text{Noiseless IVA: } \mathbf{Y}_{kt} = \mathbf{A}_k(t)\mathbf{X}_{kt} \tag{2.3}$$

$$\text{Noisy IVA: } \mathbf{Y}_{kt} = \mathbf{A}_k(t)\mathbf{X}_{kt} + \mathbf{N}_{kt}, \tag{2.4}$$

where $\mathbf{A}_k(t)$ is frequency domain response function corresponding to $h_{ij}[t]$. The $\mathbf{A}_k(t)$ is called the mixing matrix because it mixes the sources. Its inverse, $\mathbf{W}_k(t) = \mathbf{A}_k^{-1}(t)$, is called unmixing matrix, which separates the mixed signals. Figure 1 shows the mixture model of IVA.

2.2 Probabilistic Models for Source Priors. Because of the complexity of human speech production, for which there are no simple models (Ephraim & Cohen, 2006), speech is often characterized with flexible statistical models. For example, a common probability density function (PDF) for speech is a GMM, which can approximate any continuous distributions with appropriate parameters (Bishop, 1995). Because the samples are assumed to be independent and identically distributed, we drop the time index t for simplicity.

Assuming the sources are statistically independent,

$$p(\mathbf{X}_1, \ldots, \mathbf{X}_K) = \prod_{j=1}^{2} p(X_{j1}, \ldots, X_{jK})$$

$$p(X_{j1}, \ldots, X_{jK}) = \sum_{s_j} p(s_j) \prod_k \mathcal{N}(X_{jk}|0, \nu_{ks_j}). \tag{2.5}$$

The s_j indexes the mixture components of the GMM prior for source j. The gaussian PDF

$$\mathcal{N}(X_{jk}|0, \nu_{ks_j}) = \frac{\nu_{ks_j}}{\pi} e^{-\nu_{ks_j}|X_{jk}|^2} \tag{2.6}$$

is of the complex variables X_{jk}. The precision, defined as the inverse of the covariance, satisfies $1/\nu_{ks_j} = E\{|X_{jk}|^2|s_j\}$.

Consider the vector of frequency components from the same source j, $\{X_{j1}, \ldots, X_{jK}\}$. Note that although the GMM has a diagonal precision matrix for each state, the joint PDF $p(X_{j1}, \ldots, X_{jK})$ does not factorize, that is, the interdependency among the components of a vector of the same source is captured. However, the vectors originating from different sources are independent. This model, called independent vector analysis (IVA), has the advantage over ICA that the interfrequency dependency prevents permutations. All the frequency bins are separated in a correlated manner rather than separately as in ICA.

For noisy IVA, we assume a gaussian noise with precision γ,

$$p(\mathbf{Y}_k|\mathbf{X}_k) = \frac{\gamma_k^2}{\pi^2} e^{-\gamma_k|\mathbf{Y}_k - \mathbf{A}_k\mathbf{X}_k|^2}, \tag{2.7}$$

where we assume the two channels have the same noise level.

The full joint probability is given by

$$p(\mathbf{Y}_1, \ldots, \mathbf{Y}_K, \mathbf{X}_1, \ldots, \mathbf{X}_K, \mathbf{s}) = \prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{X}_k) \prod_{j=1}^{2} \left(\prod_k p(X_{jk}|s_j) p(s_j) \right), \tag{2.8}$$

where $\mathbf{s} = (s_1, s_2)$ is the collective mixture index for both sources.

The source priors can be trained in advance or estimated directly from the mixed observations. The mixing matrices $\mathbf{A}_k(t)$ and the noise spectrum γ_k are estimated from the mixed observations using an EM algorithm described later. Separated signals are constructed by applying the separation matrix to the mixed signals for the noiseless case or using minimum mean square error (MMSE) estimator for the noisy case.

2.3 Comparison to Previous Works. The original IVA (Kim et al., 2007) employed a multivariate Laplacian distribution for the source priors,

$$p(X_{j1}, \ldots, X_{jK}) \propto e^{-\sqrt{\|X_{j1}\|^2 + \cdots + \|X_{jK}\|^2}}, \tag{2.9}$$

which captures the supergaussian property of speech. This joint PDF captures the dependencies among frequency bins from the same source, thus preventing the permutation. However, this approach has some limitations. First, it uses the same PDF for all sources and is hard to adapt to different types of sources, like speech and music. Second, it is symmetric over all the frequency bins. As a result, the marginal distribution for each frequency k, $p(X_k)$ is identical. In contrast, the real sources are likely to have different cross-frequency bins for statistics. Third, it is hard to include the sensor noise.

In Moulines et al. (1997) and Attias (1999), each independent component is modeled by different GMMs. One difficulty is that the total number of mixtures grows exponentially in the number of sources. If each frequency bin has m mixtures, the joint PDF over K frequency bins contains m^K mixtures. Applying these models directly in the frequency domain is computationally intractable. A variational approximation is derived for IFA (Attias, 1999) to handle a large number of sources. Modeling each frequency bin by a GMM does not capture the interfrequency bin dependencies, and permutation correction is necessary prior to the source reconstruction.

Our IVA model has the advantages of both previous models. When a GMM is used for the joint PDF in the frequency domain, the interfrequency dependency is preserved, and permutation is prevented. The GMM models the joint PDF for a small number of mixtures and thus avoids the computational intractability of IFA. In contrast to multivariate Laplacian models, the GMM source prior can adapt to each source and separate different types of signals, such as speech and music. Further, the sensor noise can be easily handled, and the IVA can suppress noise and enhance source quality together with source separation.

3 Independent Vector Analysis for the Noiseless Case: Batch Algorithm

When the sensor noise is absent, the mixing process is given by equation 2.3:

$$\mathbf{Y}_{kt} = \mathbf{A}_k \mathbf{X}_{kt}. \tag{3.1}$$

The parameters $\theta = \{\mathbf{A}_k, v_{ks_j}, p(s_j)\}$ are estimated by maximum likelihood using the EM algorithm.

3.1 Prewhitening and Unitary Mixing and Unmixing Matrices. The scaling of $\mathbf{X}_{kt}$ and $\mathbf{A}_k$ in equation 3.1 cannot be uniquely determined by observations $\mathbf{Y}_{kt}$. Thus we can prewhiten the observations,

$$\mathbf{Q}_k = \sum_{t=0}^{T} \mathbf{Y}_{kt}\mathbf{Y}_{kt}^{\dagger} \tag{3.2}$$

$$\mathbf{Y}_{kt} \leftarrow \mathbf{Q}_k^{-\frac{1}{2}}\mathbf{Y}_{kt}, \tag{3.3}$$

where † denotes the Hermitian (complex conjugate transpose). The whitening process removes the second-order correlation, and $\mathbf{Y}_k$ has an identity covariance matrix, which facilitates the separation.

To be consistent with this whitening processes, we assume the priors are also white: $E\{|X_k|^2\} = 1$. The speech priors capture the high-order statistics of the sources, which enables IVA to achieve source separation.

It is more convenient to work with the demixing matrix defined as $\mathbf{W}_k = \mathbf{A}_k^{-1}$. Because of the prewhitening process, both mixing matrix $\mathbf{A}_k$ and demixing matrix $\mathbf{W}_k$ are unitary: $\mathbf{I} = E\{\mathbf{Y}_{kt}\mathbf{Y}_{kt}^{\dagger}\} = E\{\mathbf{A}_k\mathbf{X}_{kt}\mathbf{X}_{kt}^{\dagger}\mathbf{A}_k^{\dagger}\} = \mathbf{A}_k\mathbf{A}_k^{\dagger}$. The inverse of unitary matrix is also unitary.

We consider two sources and two microphones, and the 2×2 unitary matrix $\mathbf{W}_k$ has the Cayley-Klein parameterization

$$\mathbf{W}_k = \begin{pmatrix} a_k & b_k \\ -b_k^* & a_k^* \end{pmatrix} \quad \text{s.t. } a_k a_k^* + b_k b_k^* = 1. \tag{3.4}$$

3.2 The Expectation-Maximization Algorithm. The log-likelihood function is

$$\begin{aligned} \mathcal{L}(\theta) &= \sum_{t=1}^{T} \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) \\ &= \sum_{t=1}^{T} \log \left(\sum_{\mathbf{s}_t} \prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t) \right), \end{aligned} \tag{3.5}$$

where $\theta = \{\mathbf{W}_k, v_{ks_j}, p(s_j)\}$ consists of the model parameters, $\mathbf{s}_t = \{s_1, s_2\}$ is the collective mixture index of the GMMs for source priors, $\mathbf{Y}$ is the FFT coefficients of the mixed signal, and $p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt})$ is the PDF of the mixed signal, which is a GMM resulting from the GMM source priors.

The model parameters $\theta = \{\mathbf{W}_k, v_{ks_j}, p(s_j)\}$ are estimated by maximizing the log-likelihood function $\mathcal{L}(\theta)$, which can be done efficiently using an

EM algorithm. One appealing property of the EM algorithm is that the cost function $\mathcal{F}$ always increases. This property can be used to monitor convergence.

The detailed derivation of the EM algorithm is given in appendix A.

3.3 Postprocessing for Spectral Compensation. Because the estimated signal $\hat{\mathbf{X}}_{kt} = \mathbf{W}_k \hat{\mathbf{Y}}_{kt}$ has a flat spectrum inherited from the whitening processes, it is not appropriate for signal reconstruction, and the signal spectrum needs scaling corrections.

Let $\mathbf{X}^o_{kt}$ denote the original sources without whitening and $\mathbf{A}^o_k$ denote the real mixing matrix. The whitened mixed signal satisfies both $\mathbf{Y}_{kt} = \mathbf{Q}_k^{-1/2}\mathbf{A}^o_k\mathbf{X}^o_{kt}$ and $\mathbf{Y}_{kt} = \mathbf{A}_k\hat{\mathbf{X}}_{kt}$. Thus, $\hat{\mathbf{X}}_{kt} = \mathbf{D}_k\mathbf{X}^o_{kt}$, where $\mathbf{D}_k = \mathbf{A}_k^{-1}\mathbf{Q}_k^{-1/2}\mathbf{A}^o_k$. Recall that the components of $\hat{\mathbf{X}}_{kt}$ and $\mathbf{X}^o_{kt}$ are independent; $\hat{\mathbf{X}}_{kt}$ must be the scaled version of $\mathbf{X}^o_{kt}$ because the IVA prevents the permutations, that is, the matrix $\mathbf{D}_k$ is diagonal. Thus,

$$\operatorname{diag}(\mathbf{A}^o_k)\mathbf{X}^o_{kt} = \operatorname{diag}(\mathbf{Q}_k^{1/2}\mathbf{A}_k\mathbf{D}_k)\mathbf{X}^o_{kt} = \operatorname{diag}(\mathbf{Q}_k^{1/2}\mathbf{A}_k)\hat{\mathbf{X}}_{kt}, \tag{3.6}$$

where "diag" takes the diagonal elements of a matrix. This commutes with the diagonal matrix $\mathbf{D}_k$. We term the matrix $\operatorname{diag}(\mathbf{Q}_k^{1/2}\mathbf{A}_k)$ the spectrum compensation operator, which compensates the estimated spectrum $\hat{\mathbf{X}}_{kt}$,

$$\tilde{\mathbf{X}}_{kt} = \operatorname{diag}\left(\mathbf{Q}_k^{1/2}\mathbf{W}_k^{-1}\right)\hat{\mathbf{X}}_{kt}. \tag{3.7}$$

Note that the separated signals are filtered by $\operatorname{diag}(\mathbf{A}^o_k)$ and could suffer from reverberations. The estimated signals can be considered the recorded version of the original sources. After applying the inverse FFT to $\tilde{\mathbf{X}}_{kt}$, the time domain signals can be constructed by overlap adding, if some window is applied.

4 Independent Vector Analysis for the Noiseless Case: Online Algorithm

Under the dynamic environment, the mixing process in equation 2.3 will be time dependent:

$$\mathbf{Y}_{kt} = \mathbf{A}_k(t)\mathbf{X}_{kt}. \tag{4.1}$$

At time t, the model parameters are denoted by $\theta = \{\mathbf{A}_{kt}(t), v_{ks_j}, p(s_j)\}$, which are estimated sequentially by maximum likelihood using the EM algorithm.

4.1 Prewhitening and Unitary Mixing and Unmixing Matrices. The whitening matrices $\mathbf{Q}_k(\bar{t})$ are computed sequentially,

$$\mathbf{Q}_k(\bar{t}) = (1-\lambda)\sum_{t=0}^{\bar{t}} \lambda^{\bar{t}-t}\mathbf{Y}_{kt}\mathbf{Y}_{kt}^{\dagger} \approx \lambda\mathbf{Q}_k(\bar{t}-1) + (1-\lambda)\mathbf{Y}_{k\bar{t}}\mathbf{Y}_{k\bar{t}}^{\dagger} \tag{4.2}$$

$$\mathbf{Y}_{kt} \leftarrow \mathbf{Q}_k(\bar{t})^{-\frac{1}{2}}\mathbf{Y}_{kt}, \tag{4.3}$$

where λ is a parameter close to 1 for the online learning rate, which we explain later. The $\mathbf{Q}_k(\bar{t})$ is updated when the new sample $\mathbf{Y}_{k\bar{t}}$ is available.

As explained in the previous section, after whitening, the separation matrices are unitary and described by the Cayley-Klein parameterization

$$\mathbf{W}_k(\bar{t}) = \begin{pmatrix} a_{k\bar{t}} & b_{k\bar{t}} \\ -b_{k\bar{t}}^* & a_{k\bar{t}}^* \end{pmatrix} \quad \text{s.t. } a_{k\bar{t}}a_{k\bar{t}}^* + b_{k\bar{t}}b_{k\bar{t}}^* = 1. \tag{4.4}$$

4.2 The Expectation-Maximization Algorithm. In contrast to the batch algorithm, we consider a weighted log-likelihood function:

$$\begin{aligned}\mathcal{L}(\theta) &= \sum_{t=1}^{T} \lambda^{T-t} \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) \\ &= \sum_{t=1}^{T} \lambda^{T-t} \log \left(\sum_{\mathbf{s}_t} \prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t)p(\mathbf{s}_t) \right).\end{aligned} \tag{4.5}$$

For $0 \le \lambda \le 1$, the past samples are weighted less, and the recent samples are weighted more. The regular likelihood is obtained when $\lambda = 1$.

The model parameters θ are estimated by maximizing the weighted log-likelihood function $\mathcal{L}(\theta)$, using an EM algorithm. The variables in the E-step and M-step are updated only by the most current sample, using the proper weights corresponding to λ. This sequential updates enable the separation to adapt to the dynamic environment and the efficient online algorithm to work in real time.

The detailed derivation of the EM algorithm is given in appendix B.

4.3 Postprocessing for Spectral Compensation. Similar to the batch algorithm, the estimated signal needs spectral compensation, which can be done as

$$\tilde{\mathbf{X}}_{k\bar{t}} = \operatorname{diag}\left(\mathbf{Q}_k(\bar{t})^{1/2}\mathbf{W}_k^{-1}(\bar{t})\right)\hat{\mathbf{X}}_{k\bar{t}}. \tag{4.6}$$

After the inverse FFT is applied to $\tilde{\mathbf{X}}_{kt}$, the time domain signals can be constructed by overlap adding if some window is applied.

5 Independent Vector Analysis for the Noisy Case

When the sensor noise N_{kt} exists, the mixing process is given in equation 2.4:

$$\mathbf{Y}_{kt} = \mathbf{A}_k \mathbf{X}_{kt} + \mathbf{N}_{kt}. \tag{5.1}$$

The parameters $\theta = \{A_k, \nu_{ks_j}, p(s_j), \gamma_k\}$ are estimated by maximum likelihood using the EM algorithm. If the priors for some sources are pretrained, their corresponding parameters $\{\nu_{ks_j}, p(s_j)\}$ are fixed.

5.1 Mixing and Unmixing Matrices Are Not Unitary. The mixing matrices $\mathbf{A}_k$ are not unitary because of noise. The channel noise was assumed to be uncorrelated, but the whitening process causes the noise to become correlated, which is difficult to model and learn. For noisy IVA, the mixed signals are not prewhitened, and the mixing and unmixing matrices are not assumed to be unitary. Empirically initializing $\mathbf{A}_k$ to be the whitening matrix was suboptimal. Because the singular valuation decomposition (SVD) using Matlab gave the eigenvalues in decreasing order, the initialization with SVD would assign the frequency components with larger variances to source 1 and those with smaller variances to source 2, leading to an initial permutation bias. Thus we simply initialized $\mathbf{A}_k$ to be the identity matrix.

5.2 The Expectation-Maximization Algorithm. Again we consider the log-likelihood function as the cost

$$\begin{aligned}\mathcal{L}(\theta) &= \sum_{t=1}^{T} \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) \\ &= \sum_{t} \log \left(\sum_{\mathbf{s}_t = (s_{1t}, s_{2t})} \prod_{k=1}^{K} \int p(\mathbf{Y}_{kt}, \mathbf{X}_{kt} | \mathbf{s}_t) p(\mathbf{s}_t) d\mathbf{X}_{kt} \right).\end{aligned} \tag{5.2}$$

An EM algorithm that learns the parameters by maximizing the cost $\mathcal{L}(\theta)$ is presented in appendix C.

5.3 Signal Estimation and Spectral Compensation. Unlike the noiseless case, the signal estimation is nonlinear. The MMSE estimator is

$$\hat{\mathbf{X}}_{kt} = \sum_{\mathbf{s}_t} q(\mathbf{s}_t) \boldsymbol{\mu}_{kt\mathbf{s}_t}, \tag{5.3}$$

which is the average of the means $\boldsymbol{\mu}_{kt\mathbf{s}_t}$ weighted by the posterior state probability.

Because the estimated signal $\hat{\mathbf{X}}_{kt}$ had a flat spectrum and was not appropriate for signal reconstruction, it needed scaling correction. Let $\mathbf{X}^o_{kt}$ denote the original sources without whitening and $\mathbf{A}^o_{kt}$ denote the real mixing matrix. Under the small noise assumption, the mixed signal satisfies both $\mathbf{Y}_{kt} = \mathbf{A}^o_{kt}\mathbf{X}^o_{kt}$ and $\mathbf{Y}_{kt} = \mathbf{A}_{kt}\hat{\mathbf{X}}_{kt}$. Thus, $\hat{\mathbf{X}}_{kt} = \mathbf{D}_{kt}\mathbf{X}^o_{kt}$, where $\mathbf{D}_{kt} = \mathbf{A}^{-1}_{kt}\mathbf{A}^o_{kt}$. Recall that the components of $\hat{\mathbf{X}}_{kt}$ and $\mathbf{X}^o_{kt}$ were independent, so $\hat{\mathbf{X}}_{kt}$ must be the scaled version of $\mathbf{X}^o_{kt}$ because the IVA prevents permutations, that is, the matrix $\mathbf{D}_{kt}$ has to be diagonal. Thus,

$$\operatorname{diag}(\mathbf{A}^o_{kt})\mathbf{X}^o_{kt} = \operatorname{diag}(\mathbf{A}_{kt}\mathbf{D}_{kt})\mathbf{X}^o_{kt} = \operatorname{diag}(\mathbf{A}_{kt})\hat{\mathbf{X}}_{kt}, \tag{5.4}$$

where "diag" takes the diagonal elements of a matrix that commutes with the diagonal matrix $\mathbf{D}_{kt}$. We term the matrix $\operatorname{diag}(\mathbf{A}_{kt})$ the spectrum compensation operator, which compensates the estimated spectrum $\hat{\mathbf{X}}_{kt}$:

$$\tilde{\mathbf{X}}_{kt} = \operatorname{diag}(\mathbf{A}_{kt})\,\hat{\mathbf{X}}_{kt}. \tag{5.5}$$

Note the separated signals are filtered by $\operatorname{diag}(\mathbf{A}^o_{kt})$ and could suffer from reverberations. The estimated signals can be considered as the recorded version of the original sources. After the inverse FFT is applied on $\tilde{\mathbf{X}}_{kt}$, time domain signals can be constructed by overlap adding if some window is applied.

5.4 On the Convergence and the Online Algorithm. The mixing process reduces to a noiseless case in the limit of zero noise. Contrary to intuition, the EM algorithm for estimating the mixing matrices will not reduce to the noiseless case. The convergence is slow when the noise level is low because the update rule for $\mathbf{A}_k$ depends on the precision of noise. Petersen, Winther, and Hansen (2005) have shown that the Taylor expansion of the learning rule is

$$\mathbf{A}_k \leftarrow \mathbf{A}_k + \frac{1}{\gamma_k}\tilde{\mathbf{A}}_k + \mathcal{O}\left(\frac{1}{\gamma_k^2}\right). \tag{5.6}$$

Thus the learning rate is zero when the noise goes to zero—$\gamma_k = \infty$; essentially, $\mathbf{A}_k$ will not be updated. For this reason, the EM algorithm for noiseless IVA is derived in section 4.

In principle, we can derive an online algorithm for the noisy case in a manner similar to the noiseless case. All the variables needed for the EM algorithm can be computed recursively. Thus, the parameters of the source priors and the mixing matrices can be updated online. However, an online algorithm for the noisy case is difficult because the speed of convergence depends on the precision of noise as well as the learning rate λ we used in section 4.

6 Experimental Results for Source Separation with IVA

We demonstrate the performance of the proposed algorithm by using it to separate speech from music. Music and speech have different statistical properties, which pose difficulties for IVA using identical source priors.

6.1 Data Set Description. The music signal is a disco with a singer's voice. It is about 4.5 minutes long and sampled as 8*k* Hz. The speech signal is a male voice downloaded from the University of Florida audio news. It is about 7.5 minutes long and sampled at 8*k* Hz. These two sources were mixed together, and the task was to separate them. In the noisy IVA case, a gaussian noise at 10 dB is added to the mixtures. The goal was to suppress the noise as well as separate the signals.

Due to the flexibility of our model, it cannot learn the separation matrices and source priors from random initialization. Thus, we used the first 2 minutes of signals to train the GMM as an initialization, which was done using the standard EM algorithm (Bishop, 1995). First, a Hanning window of 1024 samples with a 50% overlap was applied to the time domain signals. Then FFT was performed on each frame. Due to the symmetry of the FFT, only the first 512 components are kept; the rest provide no additional information. The next 30 seconds of the recordings were used to evaluate the algorithms.

The 30-second-long mixed signals were obtained by simulating impulse responses of a rectangular room based on the image model technique (Allen & Berkley, 1979; Stephens & Bate, 1966; Gardner, 1992). The geometry of the room is shown in Figure 2. The reverberation time was 100 milliseconds. Similarly, a 1024-point Hanning window with 50% overlap was applied, and the FFT was used on each frame to extract the frequency components. The mixed signals in the frequency domain were processed by the proposed algorithms, as well as the benchmark algorithms.

6.2 Benchmark: Independent Vector Analysis with Laplacian Prior. The independent vector analysis was originally proposed in Kim et al. (2007), where the joint distribution of the frequency bins was assumed to be a multivariate Laplacian:

$$p(X_{j1}, \ldots, X_{jK}) \propto e^{-\sqrt{|X_{j1}|^2 + \cdots + |X_{jK}|^2}}. \tag{6.1}$$

This IVA models assumed no noise. As a result, the unmixing matrix $\mathbf{W}_k$ could be assumed to be unitary, because the mixed signals were prewhitened and estimated by maximum likelihood, defined as

$$\begin{aligned}\mathcal{L} &= \sum_t \log p(X_{11t}, \ldots, X_{1Kt}) + \log p(X_{21t}, \ldots, X_{2Kt}) \\ &= -\sum_t \sqrt{\sum_k |X_{1kt}|^2} - \sum_t \sqrt{\sum_k |X_{2kt}|^2} + c,\end{aligned} \tag{6.2}$$

where c is a constant and $\mathbf{X}_{kt} = (X_{1kt}; X_{2kt})$ is computed as $\mathbf{X}_{kt} = \mathbf{W}_k \mathbf{Y}_{kt}$.

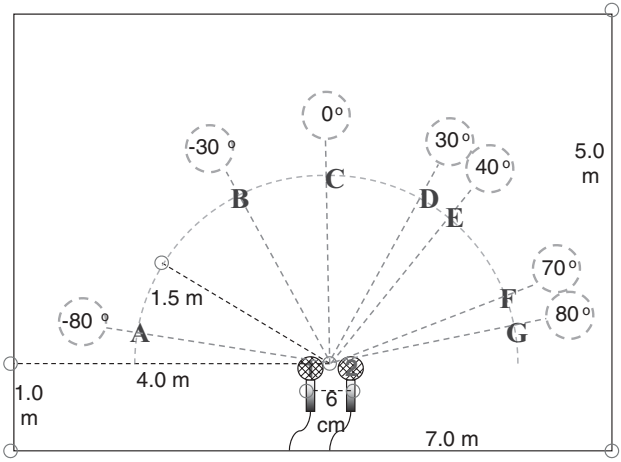

Figure 2: The size of the room is 7 m × 5 m × 2.75 m. The distance between two microphones is 6 cm. The sources are 1.5 m away from the microphones. The heights of all sources and microphones are 1.5 m. The letters (A–G) indicate the position of sources.

Optimizing $\mathcal{L}$ over $\mathbf{W}_k$ was done using gradient ascent,

$$\Delta \mathbf{W}_k = \eta \frac{\partial \mathcal{L}}{\mathbf{W}_k} \tag{6.3}$$

$$= \eta \sum_t \boldsymbol{\varphi}_{kt} \mathbf{Y}_{kt}^{\dagger}, \tag{6.4}$$

where $\boldsymbol{\varphi}_{kt} = (\frac{X_{1kt}}{\sqrt{\sum_k |X_{1kt}|^2}}, \frac{X_{2kt}}{\sqrt{\sum_k |X_{2kt}|^2}})^{\dagger}$ is the derivative of the logarithm of the source prior. The natural gradient is obtained by multiplying the right-hand side by $\mathbf{W}_k^{\dagger}\mathbf{W}_k$. The update rules become

$$\Delta \mathbf{W}_k = \eta \sum_t \boldsymbol{\varphi}_{kt} \mathbf{X}_{kt}^{\dagger} \mathbf{W}_k \tag{6.5}$$

$$\mathbf{W}_k \leftarrow \left(\mathbf{W}_k \mathbf{W}_k^{\dagger}\right)^{-\frac{1}{2}} \mathbf{W}_k, \tag{6.6}$$

where η is the learning rate, and in all experiments we used $\eta = 5$. Equation 6.6 guarantees that $\mathbf{W}_k$ is unitary.

Because the mixed signals are prewhitened, the scaling of the spectrum needs correction, as done in section 3.3.

Table 1: Signal-to-Interference Ratio for Noiseless IVA for Various Source Locations.

Source Location	D,A	D,B	D,C	D,E	D,F	D,G
IVA-Lap	11.5,18.9	11.3,13.7	11.1,12.7	10.7,15.0	11.7,18.9	12.4,19.3
IVA-GMM1	17.9,20.6	17.5,13.8	16.4,12.9	16.8,17.6	19.0,19.9	20.3,20.4
IVA-GMM2	19.7,20.7	15.1,15.7	14.0,14.0	16.8,18.6	19.6,20.2	21.4,20.8

Notes: IVA-GMM the proposed IVA using GMM as source prior. IVA-Lap: benchmark with Laplacian source prior. IVA-GMM1 updates sources, while IVA-GMM2 with source prior fixed. The first number in each cell is the SIR of the speech, and the second number is the SIR of the music.

6.3 Signal-to-Interference Ratio. The signal-to-interference ratio (SIR) for source j is defined as

$$SIR_j = 10 \log \left(\frac{\sum_{tk} |[\hat{\mathcal{W}}_{kt} \mathbf{X}^o_{kt}]_{jj}|^2}{\sum_{tk} |[\hat{\mathcal{W}}_{kt} \mathbf{X}^o_{kt}]_{lj}|^2} \right) \tag{6.7}$$

$$\hat{\mathcal{W}}_{kt} = \text{diag} \left(\mathbf{Q}^{\frac{1}{2}} \hat{\mathbf{W}}_{kt} \right) \hat{\mathbf{W}}_{kt} \mathbf{Q}_{kt}^{-\frac{1}{2}} \mathbf{A}^o_{kt}, \tag{6.8}$$

where $\mathbf{X}^o_{kt}$ is the original source. The overall impulse response $\hat{\mathcal{W}}_{kt}$ consists of the real mixing matrix, $\mathbf{A}^o_{kt}$, obtained by performing FFT on the time domain impulse response $h_{ij}[t]$, the whitening matrix, $\mathbf{Q}_{kt}^{-\frac{1}{2}}$, the separation matrix, $\hat{\mathbf{W}}_{kt}$, estimated by the EM algorithm, and the spectrum compensation, diag$(\mathbf{Q}^{\frac{1}{2}} \hat{\mathbf{W}}_{kt})$. The numerator in equation 6.7 takes the power of the estimated signal j, which is on the diagonal. The denominator in equation 6.7 computes the power of the interference, which is on the off-diagonal, $l \neq j$. Note that the permutation is prevented by IVA, and its correction is not needed.

6.4 Results for Noiseless IVA. The noiseless IVA optimizes the likelihood using the EM algorithm (see Table 1). It is guaranteed to increase the cost function, which can be used to monitor convergence. The mixed signal is whitened, and the unmixing matrices are initialized to be identity. The number of mixture for the GMM prior was 15. The GMM with 15 states was sufficient to model the joint distribution of FFT coefficients and captured their dependency. The IVA ran 12 EM iterations to learn the separation matrix with the GMM fixed. Then all the parameters were estimated from the mixtures. The convergence was very fast, taking fewer than 50 iterations, at about 1 second for each iteration. In contrast, the IVA with a Laplacian prior took around 300 iterations to converge. The speech source was placed at 30 degrees, and the music was placed at several positions. The proposed IVA-GMM improved the SIR of the speech, compared to the IVA with a Laplacian prior, IVA-Lap. Because the disco music is a mixture

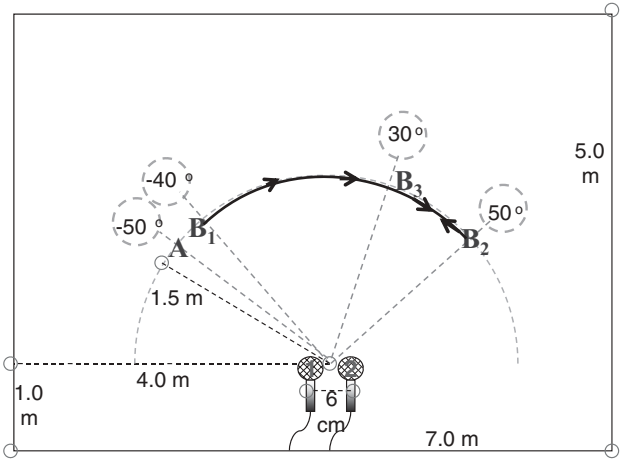

Figure 3: The speech is fixed at position A, and the music moves from B_1 to B_2 and back to B_3 at speed of 1 degree per second.

of many instruments and is a more gaussian signal due to the central limit theorem, the Laplacian distribution cannot model the music accurately. As a result, the music signal leaks into the speech channel and degrades the SIR of speech. The proposed IVA use a GMM to model music, which is more accurate than Laplacian. Thus, it prevented music from leaking into speech and improved the separation by 5 to 8 dB SIR. However, the improvement of the music is not significant because both properly model the speech and prevent it from leaking into music.

6.5 Results for Online Noiseless IVA. We applied the online IVA algorithm to separate nonstationary mixtures. The speech was fixed at location -50 degrees. The musical source was initially located at -40 degrees and moved to 50 degrees at a constant speed of 1 degree per second and then moved backward at the same speed to 20 degrees. Figure 3 shows the trajectory of the source: $B_1 \rightarrow B_2 \rightarrow B_3$.

We set the weight $\lambda = 0.95$ in our experiment, which corresponds roughly to a 5% change in the statistics for each sample. A λ that is too small overfits the recent samples, and a value that is too large slows the adaption. The choice of $\lambda = 0.95$ provided good adaption as well as reliable source separation. We trained a GMM with 15 states using the first 2 minutes of original signals, which was used to initialize the source priors of the online algorithm. The unmixing matrices were initialized to be identity. The number of EM iterations for the online algorithm is set to 1. Running more than one iteration was ineffective because the state probability computed in the E-step changes very little when the parameters are changed by one sample. The output SIR for speech and music is shown in Figures 4

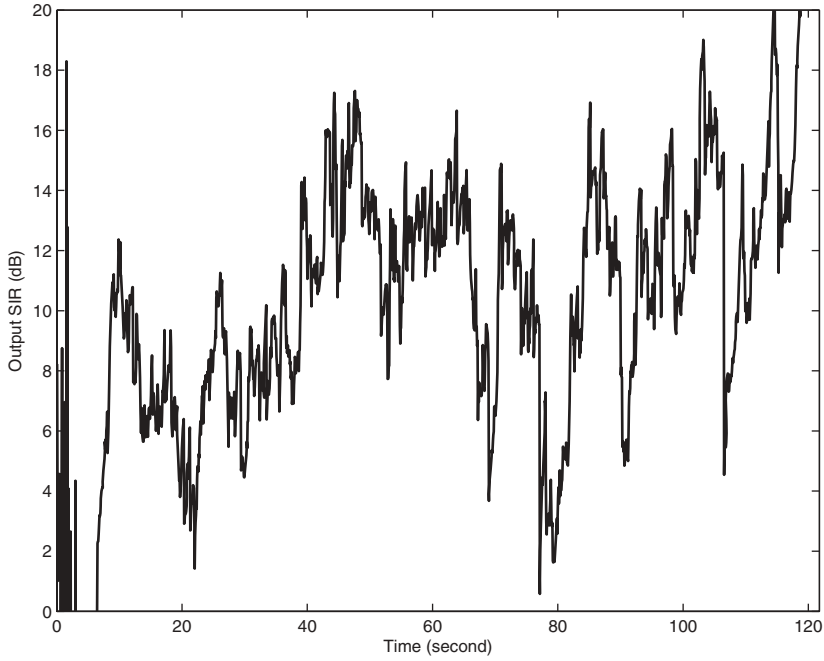

Figure 4: Output SIR for speech separated by online IVA algorithm. The speech is fixed at -50 degrees, and music moves from B_1 to B_2 and back to B_3 as indicated in Figure 3 at a speed of 1 degree per second.

and 5, respectively. The beginning period has low SIR values. The reason is due to the adaptation processes. The statistics for the beginning period were not estimated accurately, and the separation performance was low for the first 10 seconds. The SIR improved as more samples were available and the sources were separated after 10 seconds. The SIRs for both speech and music were computed locally using the unmixing matrix for each frame and 5 seconds of original signals. The silent period of speech had very low energy, which decreased the SIR. The drops of the SIR in Figure 4 corresponded to the silences in the speech singles. The output SIR for the disco music was more consistent than that of speech. However, there was a drop of the SIR for both speech and music at around 80 seconds, when the singer's voice reached a climax in disco music and confused the IVA with the human speech; SIRs for both music and speech decreased. At the end, 110 seconds, the music faded out, the SIR of speech increased and that of music decreased dramatically. The improved SIRs demonstrated that the online IVA algorithm can track the movement of the source and separate them.

6.6 Results for Noisy IVA. For the noisy case, the signals were mixed using the image method as in the noiseless case, and 10 dB white noise was

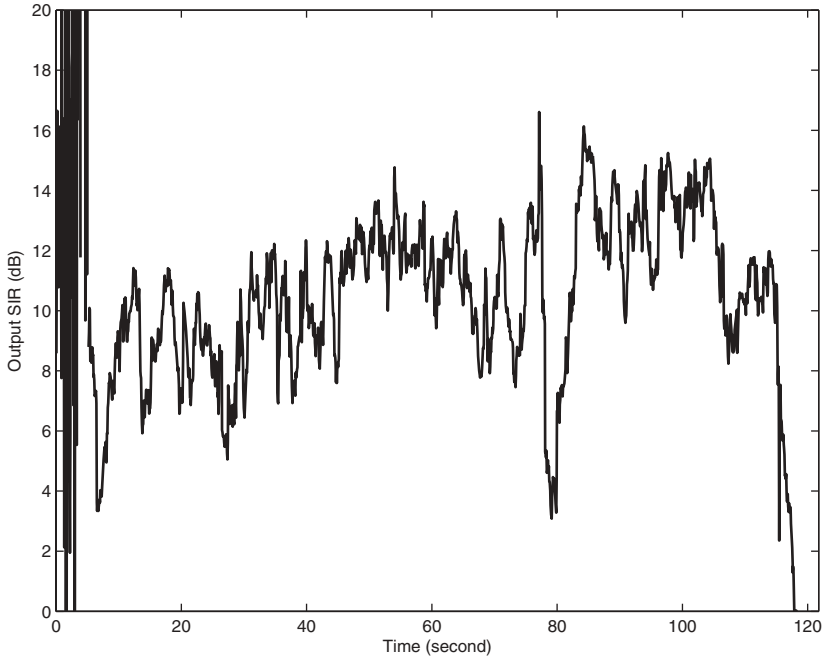

Figure 5: Output SIR for music separated by online IVA algorithm. The speech is fixed at -50 degrees, and music moves from B_1 to B_2 and back to B_3 as indicated in Figure 3 at a speed of 1 degree per second.

Table 2: Signal-to-Interference Ratio for Noisy IVA, Various Source Locations.

Source Location	D,A	D,B	D,C	D,E	D,F	D,G
IVA-GMM2	20.8,17.9	11.7,11.7	8.4,8.5	13.5,9.9	19.8,17.0	16.0,19.5

Notes: The source priors were estimated. The first number in each cell is the SIR of the speech, and the second number is the SIR of the music.

added to the mixed signals. The GMM had 15 states and was initialized by training on the first 2 minutes of the signals, with 30 seconds used for testing. The EM algorithm underwent 250 iterations, each lasting about 2 seconds. The convergence rate was slower than in the noiseless case because of the low noise. The SIRs, shown in Table 2, were close to those of the noiseless case for both speech and music, which demonstrates the effectiveness of the separation. The noise was effectively reduced, and the separated signals sounded noise free. Compared to the noiseless case, the separated signals contained no interference because the denoising process removed the interference as well as the noise. However, they had more noticeable reverberation. The reason is that the unmixing matrices were not assumed

to be unitary. The lack of regularization of the unmixing matrices made the algorithm more prone to local optima. Note that the source estimation of the IVA-GMM was nonlinear, since the state probability also depended on the observations. For nonlinear estimation, SIR may not provide a fair comparison. The spectrum compensation is not exact because of the noise, and as a result, the SIRs decreased a little compared to the noiseless case.

7 Conclusion

A novel probabilistic framework for independent vector analysis (IVA) was explored that supported EM algorithms for the noiseless case, the noisy case, and the online learning. Each source was modeled by a different GMM, which allowed different types of signals to be separated. For the noiseless case, the derived EM algorithm was rigorous, converged rapidly, and effectively separated speech and music. A general weighted likelihood cost function was used to derive an online learning algorithm for the moving sources. The parameters were updated sequentially using only the most recent sample. This adaptation process allowed the source to be tracked and separated online, which is necessary in nonstationary environments. Finally, a noisy IVA algorithm was developed that could both separate the signals and reduce the noise. Speech and music were separated based on improved SIR under the ideal conditions used in the tests. This model can also be applied to the source extraction problem. For example, to extract speech, a GMM prior can be pretrained for the speech signal, and another GMM can be used to model the interfering sources.

The formalism introduced here is quite general, and source priors other than GMM could also be used, such as the student-t distribution. However, the parameters of these distributions would have to be estimated with alternative optimization approaches rather than the efficient EM algorithm.

Appendix A: The EM Algorithm for Noiseless IVA: Batch Algorithm

Rewrite the likelihood function in equation 3.5 as

$$\begin{aligned}\mathcal{L}(\theta) &= \sum_{t=1}^{T} \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) \\ &= \sum_{t=1}^{T} \log \left(\sum_{\mathbf{s}_t} \prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t) \right),\end{aligned}$$

where $\theta = \{\mathbf{A}_k, \nu_{ks_j}, p(s_j)\}$ consists of the model parameters, $\mathbf{s}_t = \{s_1, s_2\}$ is the collective mixture index of the GMMs for source priors, $\mathbf{Y}$ is the FFT

coefficients of the mixed signal, and $p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt})$ is the PDF of the mixed signal, which is a GMM resulting from the GMM source priors. The lower bound of $\mathcal{L}(\theta)$ is

$$\begin{aligned}\mathcal{L}(\theta) &\geq \sum_{t\mathbf{s}_t} q(\mathbf{s}_t) \log \frac{\prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t)}{q(\mathbf{s}_t)} \\ &= \mathcal{F}(q, \theta) \end{aligned} \tag{A.1}$$

for distribution $q(\mathbf{s}_t)$ due to Jensen's inequality. Note that because of the absence of noise, $\mathbf{X}_{kt}$ is determined by $\mathbf{Y}_{kt}$ and is not a hidden variable. We maximized $\mathcal{L}(\theta)$ using the EM algorithm.

The EM algorithm iteratively maximizes $\mathcal{F}(q, \theta)$ over $q(\mathbf{s}_t)$ (E-step) and over θ (M-step) until convergence.

A.1 Expectation Step. For fixed θ, the $q(\mathbf{s}_t)$ that maximizes $\mathcal{F}(q, \theta)$ satisfies

$$q(\mathbf{s}_t) = \frac{\prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t)}{p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt})}. \tag{A.2}$$

Using $\mathbf{Y}_{kt} = \mathbf{W}_k \mathbf{X}_{kt}$, we obtain

$$p(\mathbf{Y}_{kt}|\mathbf{s}_t) = p(\mathbf{X}_{kt} = \mathbf{W}_k \mathbf{Y}_{kt}|\mathbf{s}_t) = \mathcal{N}(\mathbf{Y}_{kt}|0, \mathbf{\Sigma}_{k\mathbf{s}_t}). \tag{A.3}$$

The precision matrix $\mathbf{\Sigma}_{k\mathbf{s}_t}$ is given by

$$\mathbf{\Sigma}_{k\mathbf{s}_t} = \mathbf{W}_k^{\dagger} \mathbf{\Phi}_{k\mathbf{s}_t} \mathbf{W}_k; \qquad \mathbf{\Phi}_{k\mathbf{s}_t} = \begin{pmatrix} \nu_{ks_1} & 0 \\ 0 & \nu_{ks_2} \end{pmatrix}. \tag{A.4}$$

Its determinant is $\det(\mathbf{\Sigma}_{k\mathbf{s}_t}) = \nu_{ks_1} \nu_{ks_2}$, because $\mathbf{W}_k$ is unitary.

We define the function $f(\mathbf{s}_t)$ as

$$f(\mathbf{s}_t) = \sum_k \log p(\mathbf{Y}_{kt}|\mathbf{s}_t) + \log p(\mathbf{s}_t). \tag{A.5}$$

When equation A.2 is used, $q(\mathbf{s}_t) \propto e^{f(\mathbf{s}_t)}$

$$Z_t = \sum_{\mathbf{s}_t} e^{f(\mathbf{s}_t)} \tag{A.6}$$

$$q(\mathbf{s}_t) = \frac{1}{Z_t} e^{f(\mathbf{s}_t)}. \tag{A.7}$$

A.2 Maximization Step. The parameters θ was estimated by maximizing the cost function $\mathcal{F}$.

First, we consider the maximization of $\mathcal{F}$ over $\mathbf{W}_k$ under a unitary constraint. To preserve the unitarity of $\mathbf{W}_k$, using the Cayley-Klein parameterization in equation 3.4, rewrite the precision as

$$\mathbf{\Phi}_{k\mathbf{s}_t} = \begin{pmatrix} \nu_{ks_1} - \nu_{ks_2} & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} \nu_{ks_2} & 0 \\ 0 & \nu_{ks_2} \end{pmatrix}, \tag{A.8}$$

and introduce the Lagrangian multiplier β_k. After some manipulation and ignoring the constant terms in equation A.1, the $\mathbf{W}_k$ maximize

$$\begin{aligned} &-\sum_{tk\mathbf{s}_t} \lambda^{T-t} q(\mathbf{s}_t) \left\{ (\nu_{ks_1} - \nu_{ks_2}) \mathbf{Y}_{kt}^\dagger \mathbf{W}_k^\dagger \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \mathbf{W}_k \mathbf{Y}_{kt} \right\} \\ &\quad + \beta_k (a_k a_k^* + b_k b_k^* - 1) \\ &= -\sum_{tk\mathbf{s}_t} \lambda^{T-t} q(\mathbf{s}_t) (\nu_{ks_1} - \nu_{ks_2}) |a_k Y_{1kt} + b_k Y_{2kt}|^2 + \beta_k (a_k a_k^* + b_k b_k^* - 1). \end{aligned} \tag{A.9}$$

Because this is quadratic in a_k and b_k, an analytical solution exists. When the derivatives with respect to a_k and b_k are set to zero, we have

$$\mathbf{M}_{kT} \begin{pmatrix} a_k^* \\ b_k^* \end{pmatrix} = \beta_k \begin{pmatrix} a_k^* \\ b_k^*, \end{pmatrix} \tag{A.10}$$

where $\mathbf{M}_{kT}$ is defined as

$$\mathbf{M}_{kT} = \sum_{t\mathbf{s}_t} q(\mathbf{s}_t) (\nu_{ks_1} - \nu_{ks_2}) \mathbf{Y}_{kt} \mathbf{Y}_{kt}^\dagger. \tag{A.11}$$

The vector $(a_k, b_k)^\dagger$ is the eigenvector of $\mathbf{M}_{kT}$ with a smaller eigenvalue. This can be shown as follows.

We use equation A.11 to compute the value of the objective function, equation A.9;

$$-\mathrm{Tr} \left\{ \sum_{tk\mathbf{s}_t} q(\mathbf{s}_t) (\nu_{ks_1} - \nu_{ks_2}) \mathbf{Y}_{kt} \mathbf{Y}_{kt}^\dagger \mathbf{W}_k^\dagger \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \mathbf{W}_k \right\} \tag{A.12}$$

$$= -\mathrm{Tr} \left\{ \mathbf{M}_{kT} \begin{pmatrix} a_k^* \\ b_k^* \end{pmatrix} (a_k \;\; b_k) \right\} = -\beta_k. \tag{A.13}$$

Thus, the eigenvector associated with the smaller eigenvalue gives the higher value of the cost function. Thus, $(a_k, b_k)^\dagger$ is the eigenvector of $\mathbf{M}_{kT}$ with the smaller eigenvalue.

The eigenvalue problem in equation A.10 can be solved analytically for the 2×2 case. Write $\mathbf{M}_{kT}$

$$\mathbf{M}_{kT} = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}, \tag{A.14}$$

where M_{11}, M_{22} are real and $M_{21} = M_{12}^*$, because $\mathbf{M}_{kT}$ is Hermitian. Ignore the subscript k for simplicity. Its eigenvalues are $\frac{M_{11}+M_{22}}{2} \pm \sqrt{\frac{(M_{11}-M_{22})^2}{4} + |M_{12}|^2}$, which are real. The smaller one is

$$\beta_k = \frac{M_{11} + M_{22}}{2} - \sqrt{\frac{(M_{11} - M_{22})^2}{4} + |M_{12}|^2}, \tag{A.15}$$

and the corresponding eigenvector is

$$\begin{pmatrix} a_k^* \\ b_k^* \end{pmatrix} = \frac{1}{\sqrt{1 + (\frac{\beta_k - M_{11}}{M_{12}})^2}} \begin{pmatrix} 1 \\ \frac{\beta_k - M_{11}}{M_{12}} \end{pmatrix}. \tag{A.16}$$

This analytical solution avoids complicated matrix calculations and greatly improves the efficiency.

Maximizing $\mathcal{F}(q, \theta)$ over $\{\nu_{ks_j}, p(s_j)\}$ is straightforward. For the precision ν_{ks_j}, we have

$$\frac{1}{\nu_{ks_j=r}} = \frac{\left[\sum_{t,\mathbf{s}_t} q(s_{jt} = r) \mathbf{W}_k \mathbf{Y}_{kt} \mathbf{Y}_{kt}^\dagger \mathbf{W}_k^\dagger\right]_{jj}}{\sum_{t,\mathbf{s}_t} q(s_{jt} = r)}, \tag{A.17}$$

where $[\cdot]_{jj}$ denotes the (j, j) element of the matrix. The state probability is

$$p(s_j = r) = \frac{\sum_{t=1}^{T} q(s_{jt} = r)}{T}. \tag{A.18}$$

The cost function $\mathcal{F}$ is easily accessible as a by-product of the E-step. Using equations A.5 and A.6, we have $Z_t = p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt})$ and

$$\mathcal{F}(q, \theta) = \sum_t \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) = \sum_t \log(Z_t), \tag{A.19}$$

One appealing property of the EM algorithm is that the cost function $\mathcal{F}$ always increases. This property can be used to monitor convergence. The above E-step and M-step iterate until some convergent criterion is satisfied.

Appendix B: The EM Algorithm for Noiseless IVA: Online Algorithm

The weighted log-likelihood function in equation 4.5 is

$$\begin{aligned}\mathcal{L}(\theta) &= \sum_{t=1}^{T} \lambda^{T-t} \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) \\ &= \sum_{t=1}^{T} \lambda^{T-t} \log \left(\sum_{\mathbf{s}_t} \prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t) \right) \\ &\geq \sum_{t\mathbf{s}_t} \lambda^{T-t} q(\mathbf{s}_t) \log \frac{\prod_{k=1}^{K} p(\mathbf{Y}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t)}{q(\mathbf{s}_t)} \\ &= \mathcal{F}(q, \theta) \end{aligned} \tag{B.1}$$

for distribution $q(\mathbf{s}_t)$ due to Jensen's inequality. We maximized $\mathcal{L}(\theta)$ using the EM algorithm,

B.1 Expectation Step. For fixed θ, the $q(\mathbf{s}_{\bar{t}})$ maximizes $\mathcal{F}(q, \theta)$. This step is same as the batch algorithm, except that the parameters at frame $\bar{t} - 1$ are used:

$$q(\mathbf{s}_{\bar{t}}) = \frac{\prod_{k=1}^{K} p(\mathbf{Y}_{k\bar{t}}|\mathbf{s}_{\bar{t}}) p(\mathbf{s}_{\bar{t}})}{p(\mathbf{Y}_{1\bar{t}}, \ldots, \mathbf{Y}_{K\bar{t}})}. \tag{B.2}$$

Using $\mathbf{Y}_{k\bar{t}} = \mathbf{W}_k(\bar{t} - 1)\mathbf{X}_{k\bar{t}}$, we obtain

$$p(\mathbf{Y}_{k\bar{t}}|\mathbf{s}_{\bar{t}}) = p(\mathbf{X}_{k\bar{t}} = \mathbf{W}_k(\bar{t} - 1)\mathbf{Y}_{k\bar{t}}|\mathbf{s}_{\bar{t}}) = \mathcal{N}(\mathbf{Y}_{k\bar{t}}|0, \mathbf{\Sigma}_{k\mathbf{s}_{\bar{t}}}), \tag{B.3}$$

where $\mathbf{\Sigma}_{k\mathbf{s}_{\bar{t}}}$ is given by

$$\mathbf{\Sigma}_{k\mathbf{s}_{\bar{t}}} = \mathbf{W}_k^{\dagger}(\bar{t} - 1)\mathbf{\Phi}_{k\mathbf{s}_{\bar{t}}}\mathbf{W}_k(\bar{t} - 1); \qquad \mathbf{\Phi}_{k\mathbf{s}_{\bar{t}}} = \begin{pmatrix} \nu_{ks_1} & 0 \\ 0 & \nu_{ks_2} \end{pmatrix}. \tag{B.4}$$

Its determinant is $\det(\mathbf{\Sigma}_{k\mathbf{s}_{\bar{t}}}) = \nu_{ks_1}\nu_{ks_2}$, because $\mathbf{W}_k(\bar{t} - 1)$ is unitary.

We define the function $f(\mathbf{s}_{\bar{t}})$ as

$$f(\mathbf{s}_{\bar{t}}) = \sum_{k} \log p(\mathbf{Y}_{k\bar{t}}|\mathbf{s}_{\bar{t}}) + \log p(\mathbf{s}_{\bar{t}}). \tag{B.5}$$

When equation B.2 is used, $q(\mathbf{s}_{\bar{t}}) \propto e^{f(\mathbf{s}_{\bar{t}})}$,

$$Z_{\bar{t}} = \sum_{\mathbf{s}_{\bar{t}}} e^{f(\mathbf{s}_{\bar{t}})} \tag{B.6}$$

$$q(\mathbf{s}_{\bar{t}}) = \frac{1}{Z_{\bar{t}}} e^{f(\mathbf{s}_{\bar{t}})}. \tag{B.7}$$

In contrast to the batch algorithm, where the mixture probabilities for all frames are updated, this online algorithm computes the mixture probability for only the most recent frame $\bar{t}$ using the model parameters at frame $\bar{t}-1$.

B.2 Maximization Step. We now derive an M-step that updates the parameters sequentially. As in the batch algorithm, the $(a_k(t), b_k(t))^\dagger$ is the eigenvector of $\mathbf{M}_k(\bar{t})$ with the smaller eigenvalue,

$$\mathbf{M}_k(\bar{t}) \begin{pmatrix} a_k^*(\bar{t}) \\ b_k^*(\bar{t}) \end{pmatrix} = \beta_k \begin{pmatrix} a_k^*(\bar{t}) \\ b_k^*(\bar{t}) \end{pmatrix}, \tag{B.8}$$

where $\mathbf{M}_k(\bar{t})$ is defined as

$$\mathbf{M}_k(\bar{t}) = \sum_{t\mathbf{s}_t}^{\bar{t}} \lambda^{\bar{t}-t} q(\mathbf{s}_t)(\nu_{ks_1} - \nu_{ks_2}) \mathbf{Y}_{kt} \mathbf{Y}_{kt}^\dagger \tag{B.9}$$

$$= \lambda \mathbf{M}_k(\bar{t}-1) + \sum_{\mathbf{s}_{\bar{t}}} q(\mathbf{s}_{\bar{t}})(\nu_{ks_1} - \nu_{ks_2}) \mathbf{Y}_{k\bar{t}} \mathbf{Y}_{k\bar{t}}^\dagger. \tag{B.10}$$

This matrix contains all the information for computing the separation matrices. It is updated by recent samples only. The eigenvalue and eigenvector are computed analytically using the method in equations A.15 and A.16.

To derive the update rules for $\{\nu_{ks_j}, p(s_j)\}$, we define the effective number of samples belonging to state r, up to time $\bar{t}$, for source j:

$$m_{jr}(\bar{t}) = \sum_{t=1}^{\bar{t}} \lambda^{\bar{t}-t} q(s_{jt} = r) = \lambda m_{jr}(\bar{t}-1) + q(s_{j\bar{t}} = r). \tag{B.11}$$

Then

$$\frac{1}{\nu_{ks_j=r}(\bar{t})} = \frac{\left[\sum_{t,\mathbf{s}_t} \lambda^{\bar{t}-t} q(s_{jt} = r) \mathbf{W}_k(\bar{t}) \mathbf{Y}_{kt} \mathbf{Y}_{kt}^\dagger \mathbf{W}_k(\bar{t})^\dagger\right]_{jj}}{\sum_{t,\mathbf{s}_t} \lambda^{\bar{t}-t} q(s_{jt} = r)} \tag{B.12}$$

$$= \frac{1}{\nu_{ks_j=r}(\bar{t}-1)} \frac{m_{jr}(\bar{t}-1)}{m_{jr}(\bar{t})} + \frac{q(s_{j\bar{t}} = r)}{m_{jr}(\bar{t})}$$

$$\times \left[\mathbf{W}_k(\bar{t})\mathbf{Y}_{kt}\mathbf{Y}_{kt}^{\dagger}\mathbf{W}_k(\bar{t})^{\dagger}\right]_{jj} \tag{B.13}$$

$$p(s_j = r) = \frac{\sum_{t=1}^{\bar{t}} \lambda^{\bar{t}-t} q(s_{jt} = r)}{T} = \frac{m_{jr}(\bar{t})}{\sum_r m_{jr}(\bar{t})}. \tag{B.14}$$

For the online algorithm, the variables $\mathbf{M}_k(\bar{t})$ and $m_{jr}(\bar{t})$ are computed by averaging over their previous values and the information from the new sample. The λ reflects how much weight is on the past values.

Appendix C: The EM Algorithm for Noisy IVA

The log-likelihood function in equation 5.2 is

$$\begin{aligned}
\mathcal{L}(\theta) &= \sum_{t=1}^{T} \log p(\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt}) \\
&= \sum_t \log \left(\sum_{\mathbf{s}_t=(s_{1t},s_{2t})} \prod_{k=1}^{K} \int p(\mathbf{Y}_{kt}, \mathbf{X}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t) d\mathbf{X}_{kt} \right) \\
&\geq \sum_{t\mathbf{s}_t} \int \prod_{k=1}^{K} q(\mathbf{X}_{kt}|\mathbf{s}_t) q(\mathbf{s}_t) \times \log \frac{\prod_{k=1}^{K} p(\mathbf{Y}_{kt}, \mathbf{X}_{kt}|\mathbf{s}_t) p(\mathbf{s}_t)}{\prod_{k=1}^{K} q(\mathbf{X}_{kt}|\mathbf{s}_t) q(\mathbf{s}_t)} d\mathbf{X}_{kt} \\
&= \mathcal{F}(q, \theta).
\end{aligned} \tag{C.1}$$

The inequality is due to Jensen's inequality and is valid for any PDF $q(\mathbf{X}_{kt}, \mathbf{s}_t)$. Equality $\mathcal{F} = \mathcal{L}$ occurs q equals to the posterior PDF $q(\mathbf{X}_{kt}, \mathbf{s}_t) = p(\mathbf{X}_{kt}, \mathbf{s}_t|\mathbf{Y}_{1t}, \ldots, \mathbf{Y}_{Kt})$.

C.1 Expectation Step. For fixed θ, the $q(\mathbf{X}_{kt}|\mathbf{s}_t)$ that maximizes $\mathcal{F}(q, \theta)$ satisfies

$$q(\mathbf{X}_{kt}|\mathbf{s}_t) = p(\mathbf{X}_{kt}|\mathbf{s}_t, \mathbf{Y}_{kt}) = \frac{p(\mathbf{Y}_{kt}|\mathbf{X}_{kt}) p(\mathbf{X}_{kt}|\mathbf{s}_t)}{p(\mathbf{Y}_{kt}|\mathbf{s}_t)}, \tag{C.2}$$

which is a gaussian PDF given by

$$q(\mathbf{X}_{kt}|\mathbf{s}_t) = \mathcal{N}(\mathbf{X}_{kt}|\boldsymbol{\mu}_{kt\mathbf{s}_t}, \boldsymbol{\Phi}_{k\mathbf{s}_t}) \tag{C.3}$$

$$\boldsymbol{\mu}_{kt\mathbf{s}_t} = \gamma_k \boldsymbol{\Phi}_{k\mathbf{s}_t}^{-1} \mathbf{A}_k^{\dagger} \mathbf{Y}_{kt} \tag{C.4}$$

$$\boldsymbol{\Phi}_{k\mathbf{s}_t} = \gamma_k \mathbf{A}_k^{\dagger} \mathbf{A}_k + \begin{pmatrix} v_{ks_{1t}} & 0 \\ 0 & v_{ks_{2t}} \end{pmatrix}. \tag{C.5}$$

To compute the optimal $q(\mathbf{s}_t)$, define the function $f(\mathbf{s}_t)$,

$$f(\mathbf{s}_t) = \sum_k \log p(\mathbf{Y}_{kt}|\mathbf{s}_t) + \log p(\mathbf{s}_t)$$

$$= \sum_k \left(\log \left| \frac{\mathbf{\Sigma}_{k\mathbf{s}_t}}{\pi} \right| - \mathbf{Y}_{kt}^\dagger \mathbf{\Sigma}_{k\mathbf{s}_t} \mathbf{Y}_{kt} \right) + \log p(\mathbf{s}_t), \tag{C.6}$$

where $p(\mathbf{Y}_{kt}|\mathbf{s}_t) = \int p(\mathbf{Y}_{kt}|\mathbf{X}_{kt}) p(\mathbf{X}_{kt}|\mathbf{s}_t) d\mathbf{X}_{kt} = \mathcal{N}(\mathbf{Y}_{kt}|0, \mathbf{\Sigma}_{k\mathbf{s}_t})$ and the precision $\mathbf{\Sigma}_{k\mathbf{s}_t}$ is

$$\mathbf{\Sigma}_{k\mathbf{s}_t}^{-1} = \mathbf{A}_k \begin{pmatrix} \frac{1}{\nu_{ks_1}} & 0 \\ 0 & \frac{1}{\nu_{ks_2}} \end{pmatrix} \mathbf{A}_k^\dagger + \frac{1}{\gamma_k} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{C.7}$$

Because $q(\mathbf{s}_t) \propto e^{f(\mathbf{s}_t)}$, we have

$$q(\mathbf{s}_t) = \frac{1}{Z_t} e^{f(\mathbf{s}_t)} \tag{C.8}$$

$$Z_t = \sum_{\mathbf{s}_t} e^{f(\mathbf{s}_t)}. \tag{C.9}$$

C.2 Maximization Step. The M-step maximizes the cost $\mathcal{F}$ over θ, which is achieved by setting the derivatives of $\mathcal{F}$ to zero.

Setting the derivative of $\mathcal{F}(q, \theta)$ with respect to $\mathbf{A}_k$ to zero, we obtain

$$\mathbf{A}_k \mathbf{U}_k = \mathbf{V}_k, \tag{C.10}$$

where

$$\mathbf{U}_k = \sum_{t=0}^{T} \sum_{\mathbf{s}_t} E^q \{ \mathbf{X}_{kt} \mathbf{X}_{kt}^\dagger \} \tag{C.11}$$

$$\mathbf{V}_k = \sum_{t=0}^{T} \sum_{\mathbf{s}_t} E^q \{ \mathbf{Y}_{kt} \mathbf{X}_{kt}^\dagger \}. \tag{C.12}$$

The expectations are given by

$$E^q \{ \mathbf{Y}_{kt} \mathbf{X}_{kt}^\dagger \} = \sum_{\mathbf{s}_t} q(\mathbf{s}_t) \mathbf{Y}_{kt} \boldsymbol{\mu}_{kt\mathbf{s}_t}^\dagger \tag{C.13}$$

$$= \sum_{\mathbf{s}_t} q(\mathbf{s}_t) \gamma_k \mathbf{Y}_{kt} \mathbf{Y}_{kt}^\dagger \mathbf{A}_k \mathbf{\Phi}_{k\mathbf{s}_t}^{-1} \tag{C.14}$$

$$E^q\{\mathbf{X}_{kt}\mathbf{X}_{kt}^\dagger\} = \sum_{\mathbf{s}t} q(\mathbf{s}_t)\left(\mathbf{\Phi}_{k\mathbf{s}_t}^{-1} + \boldsymbol{\mu}_{kt\mathbf{s}_t}\boldsymbol{\mu}_{kt\mathbf{s}_t}^\dagger\right) \tag{C.15}$$

$$= \sum_{\mathbf{s}t} q(\mathbf{s}_t)\left(\mathbf{\Phi}_{k\mathbf{s}_t}^{-1} + \gamma_k^2\mathbf{\Phi}_{k\mathbf{s}_t}^{-1}\mathbf{A}_k^\dagger\mathbf{Y}_{kt}\mathbf{Y}_{kt}^\dagger\mathbf{A}_k\mathbf{\Phi}_{k\mathbf{s}_t}^{-1}\right). \tag{C.16}$$

The bottleneck of the EM algorithm lies in the computation of $\boldsymbol{\mu}_{kt\mathbf{s}_t}$ in the expectation step, which is avoid by using equation C.4. Fortunately, the common terms can be computed once, and $\mathbf{Y}_{kt}\mathbf{Y}_{kt}^\dagger$ can be computed in advance.

Similarly, we can obtain the update rules for the parameters of the source j, $\{\nu_{ks_j},\ p(s_j)\}$ and the noise precision γ_k:

$$\frac{1}{\nu_{ks_j=r}} = \frac{\left[\sum_{t,\mathbf{s}_t}\delta_{rs_{jt}}q(\mathbf{s}_t)(\mathbf{\Phi}_{k\mathbf{s}_t}^{-1} + \boldsymbol{\mu}_{kt\mathbf{s}_t}\boldsymbol{\mu}_{kt\mathbf{s}_t}^\dagger)\right]_{jj}}{\sum_{t,\mathbf{s}_t}\delta_{rs_{jt}}q(\mathbf{s}_t)} \tag{C.17}$$

$$p(s_j = r) = \frac{\sum_{t,\mathbf{s}_t}\delta_{rs_{jt}}q(\mathbf{s}_t)}{\sum_{t,\mathbf{s}_t}q(\mathbf{s}_t)} = \frac{1}{T}\sum_t q(s_{jt} = r) \tag{C.18}$$

$$\frac{1}{\gamma_k} = \frac{\sum_t E^q\{(\mathbf{Y}_{kt} - \mathbf{A}_k\mathbf{X}_{kt})^\dagger(\mathbf{Y}_{kt} - \mathbf{A}_k\mathbf{X}_{kt})\}}{2T}$$

$$= \frac{1}{2T}\sum_t \mathrm{Tr}\Big[\mathbf{Y}_{kt}\mathbf{Y}_{kt}^\dagger - \mathbf{A}_k E^q\{\mathbf{X}_{kt}\mathbf{Y}_{kt}^\dagger\}$$

$$- \mathbf{A}_k^\dagger E^q\{\mathbf{Y}_{kt}\mathbf{X}_{kt}^\dagger\} + \mathbf{A}_k E^q\{\mathbf{X}_{kt}\mathbf{X}_{kt}^\dagger\}\mathbf{A}_k^\dagger\Big], \tag{C.19}$$

where the $\delta_{rs_{jt}}$ is the Kronecker delta function: $\delta_{rs_{jt}} = 1$ if $s_{jt} = r$ and $\delta_{rs_{jk}} = 0$ otherwise. Essentially the state for the source j is fixed to be r. The $[\cdot]_{jj}$ denotes the (j, j) element of the matrix. The identity $\mathbf{Y}_k^\dagger\mathbf{Y}_k = \mathrm{Tr}[\mathbf{Y}_k\mathbf{Y}_k^\dagger]$ is used. The $E^q\{\mathbf{Y}_{kt}\mathbf{X}_{kt}^\dagger\}$ is given by equation C.13, $E^q\{\mathbf{X}_{kt}\mathbf{Y}_{kt}^\dagger\} = E^q\{\mathbf{Y}_{kt}\mathbf{X}_{kt}^\dagger\}^\dagger$, and $E^q\{\mathbf{X}_{kt}\mathbf{X}_{kt}^\dagger\}$ is given by equation C.15.

Acknowledgments

We thank the anonymous reviewers for valuable comments and suggestions.

References

Allen, J. B., & Berkley, D. A. (1979). Image method for efficiently simulating small room acoustics. *J. Acoust. Soc. Amer., 65*, 943–950.

Attias, H. (1999). Independent factor analysis. *Neural Computation, 11*(5), 803–851.

Attias, H., Deng, L., Acero, A., & Platt, J. (2001). A new method for speech denoising and robust speech recognition using probabilistic models for clean speech and for noise. In *Proc. Eurospeech* (Vol. 3, pp. 1903–1906).

Attias, H., Platt, J. C., Acero, A., & Deng, L. (2000). Speech denoising and dereverberation using probabilistic models. In T. K. Leen, T. G. Dietterich, & V. Tresp (Eds.), *Advances in neural information processing systems, 13* (pp. 758–764). Cambridge, MA: MIT Press.

Bach, F. R., & Jordan, M. I. (2002). Kernel independent component analysis. *Journal of Machine Learning Research, 3*, 1–48.

Bell, A. J., & Sejnowski, T. J. (1995). An information maximization approach to blind separation and blind deconvolution. *Neural Computation, 7*(6), 1129–1159.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. New York: Oxford University Press.

Boscolo, R., Pan, H., & Roychowdhury, V. P. (2004). Independent component analysis based on nonparametric density estimation. *IEEE Trans. on Neural Networks, 15*(1), 55–65.

Cardoso, J.-F. (1999). High-order contrasts for independent component analysis. *Neural Computation, 11*(1), 157–192.

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing, 36*(3), 287–314.

Ephraim, Y., & Cohen, I. (2006). Recent advancements in speech enhancement. In R. C. Dorf (Ed.), *The electrical engineering handbook*. Boca Raton, FL: CRC Press.

Gardner, W. G. (1992). *The virtual acoustic room*. Unpublished master's thesis, MIT.

Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks, 10*(3), 626–634.

Hyvärinen, A., & Hoyer, P. O. (2000). Emergence of phase- and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation, 12*(7), 1705–1720.

Hyvärinen, A., Hoyer, P. O., & Inki, M. (2001). Topographic independent component analysis. *Neural Computation, 13*(7), 1527–1558.

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent component analysis*. New York: Wiley.

Hyvärinen, A., & Oja, E. (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation, 9*(7), 1483–1492.

Kim, T., Attias, H. T., Lee, S.-Y., & Lee, T.-W. (2007). Blind source separation exploiting higher-order frequency dependencies. *IEEE Trans. on Speech and Audio Processing, 15*(1), 70–79.

Kurita, S., Saruwatari, H., Kajita, S., Takeda, K., & Itakura, F. (2000). Evaluation of blind signal separation method using directivity pattern under reverberant conditions. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing* (pp. 3140–3143). Piscataway, NJ: IEEE Press.

Lee, I., Kim, T., & Lee, T.-W. (2007). Fast fixed-point independent vector analysis algorithms for convolutive blind source separation. *Signal Processing, 87*(8), 1859–1871.

Lee, I., & Lee, T.-W. (2007). On the assumption of spherical symmetry and sparseness for the frequency-domain speech model. *IEEE Trans. on Speech, Audio and Language Processing, 15*(5), 1521–1528.

Lee, T.-W. (1998). *Independent component analysis: Theory and applications*. Norwell, MA: Kluwer.

Lee, T.-W., Bell, A. J., & Lambert, R. (1997). Blind separation of convolved and delayed sources. In M. I. Jordan, M. Kearns, & S. A. Solla (Eds.), *Advances in neural information processing systems* (pp. 758–764). Cambridge, MA: MIT Press.

Lee, T.-W., Girolami, M., & Sejnowski, T. J. (1999). Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and supergaussian sources. *Neural Computation, 11*, 417–441.

Mitianoudis, N., & Davies, M. (2003). Audio source separation of convolutive mixtures. *IEEE Trans. on Speech and Audio Processing, 11*(5), 489–497.

Moulines, E., Cardoso, J.-F., & Gassiat, E. (1997). Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing* (pp. 3617–3620). Piscataway, NJ: IEEE Press.

Murata, N., Ikeda, S., & Ziehe, A. (2001). An approach to blind source separation based on temporal structure of speech signals. *Neurocomputing, 41*, 1–24.

Parra, L., & Spence, C. (2000). Convolutive blind separation of non-stationary sources. *IEEE Trans. on Speech and Audio Processing, 8*(3), 320–327.

Petersen, K. B., Winther, O., & Hansen, L. K. (2005). On the slow convergence of em and vbem in low-noise linear models. *Neural Computation, 17*, 1921–1926.

Smaragdis, P. (1998). Blind separation of convolved mixtures in the frequency domain. *Neurocomputing, 22*, 21–34.

Stephens, R. B., & Bate, A. E. (1966). *Acoustics and vibrational physics*. London: Edward Arnold Publishers.

Torkkola, K. (1966). Blind separation of convolved sources based on information maximization. *Proc. IEEE Int. Workshop on Neural Networks for Signal Processing* (pp. 423–432). Piscataway, NJ: IEEE Press.

Received November 7, 2008; accepted September 26, 2009.