Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2016, Article ID 3982360, 13 pages
http://dx.doi.org/10.1155/2016/3982360

Research Article

Underdetermined Separation of Speech Mixture Based on Sparse Bayesian Learning

Zhe Wang,[1] Luyun Wang,[2] Xiumei Li,[3] Lifan Zhao,[1] and Guoan Bi[1]

[1]*School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore*
[2]*School of Electronic Engineering and Automation, City College of Dalian University of Technology, Dalian, China*
[3]*School of Information Science and Engineering, Hangzhou Normal University, Hangzhou, China*

Correspondence should be addressed to Xiumei Li; lixiumei@pmail.ntu.edu.sg

Received 31 March 2016; Revised 1 September 2016; Accepted 19 September 2016

Academic Editor: Eric Feulvarch

This paper describes a novel algorithm for underdetermined speech separation problem based on compressed sensing which is an emerging technique for efficient data reconstruction. The proposed algorithm consists of two steps. The unknown mixing matrix is firstly estimated from the speech mixtures in the transform domain by using K-means clustering algorithm. In the second step, the speech sources are recovered based on an autocalibration sparse Bayesian learning algorithm for speech signal. Numerical experiments including the comparison with other sparse representation approaches are provided to show the achieved performance improvement.

1. Introduction

In recent years, compressed sensing (CS) theory [1, 2] has attracted a great deal of attention for various applications. It is a novel concept to directly sample the signals in a compressed manner and the signals in a lossless or robust manner, under the assumption that the signals have sparse or compressible representation in a particular domain [2, 3]. In particular, the sensing procedure in CS can preserve useful information embedded in the high-dimensional signals and the CS recovering procedure can robustly reconstruct the original sparse signals from these collected low-dimensional samples [3]. In this manner, both sensing and storage costs can be substantially saved. It provides potentially a powerful framework for computing a sparse representation of signals. The key factor allowing the success of the CS technique is proper exploitation and utilization of sparsity. In practical applications, fortunately, sparsity of the signal widely exists in various applications.

Speech separation refers to the process of separating source signals from their mixtures [4, 5]. When the number of mixtures is greater than or equal to the number of sources, independent component analysis (ICA) [6] based methods are widely used. However, for the case of underdetermined separation, where the number of mixtures is less than the number of sources, ICA based methods generally fail to separate the sources. In this context, the sparsity of the signal is often utilized to separate the source signals [5, 7, 8]. A signal is considered to be sparse if most of its samples are zero [4]. Since signals such as speech are more sparse in the time frequency (TF) domain compared to that in the time domain, several algorithms have been proposed for the separation of the signals in their TF domain [7–10]. In the received mixtures, a single source point is defined as any TF point that is associated with only one source signal. If all the TF points are single source points, the sources are known to be W-disjoint. Assuming W-disjoint sources, the degenerate unmixing estimation technique (DUET) [7] first estimates the feature vector consisting of TF points. The extracted feature vectors are then clustered to separate the sources.

In the underdetermined speech separation problem, the underdetermined mixture is a form of compressed sampling, and therefore CS theory can be utilized to solve the problem. The similarities between CS and source separation are shown in [11]. Xu and Wang developed a framework for this problem based on CS using fixed dictionary [12], while

they proposed a multistage method for underdetermined speech separation using block-based CS [13]. However, all these mentioned methods ignore the error brought in after calculating the mixing matrix. Different from the previously reported work, our proposed approach can be considered to be parametric and is particularly tailored to solve the speech recovery with inaccurate estimation of the mixing matrix. The problem is formulated in a sparse Bayesian framework and solved by Bayesian inference technique due to the privileges of the sparse Bayesian algorithm [14–16]. It operates in a statistical alternating fashion, where both the estimation and uncertainty information are utilized. Moreover, for calibrating the inaccurate mixing matrix, this framework facilitates parameter learning procedures.

The rest of the paper is organized as follows. In Section 2, the underdetermined speech separation problem is formulated into a compressed sensing framework. In Sections 3 and 4, our sparse Bayesian algorithm is proposed and speech recovery algorithm which deals with both mixing error and speech recovery is described. Numerical experiments and conclusion are given in Sections 5 and 6, respectively.

2. The CS Framework of Underdetermined Separation

The task of speech separation is to recover the sources using the observable signals. The noise-free instantaneous mixing model can be described as follows:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t), \tag{1}$$

where the mixing matrix $\mathbf{A} \in R^{(M\times N)}$ is unknown, $\mathbf{x}(t) \in R^{(M)}$ is the observed data vector at discrete time instant t, $\mathbf{s}(t) \in R^{(N)}$ is the unknown source vector, M is the number of the microphones, and N is the number of the sources. In this paper, we focus on the underdetermined speech separation; that is, $M < N$.

Let us expand (1) as

$$\begin{bmatrix} x_1(t) \\ \vdots \\ x_M(t) \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \vdots & \vdots \\ a_{M1} & \cdots & a_{MN} \end{bmatrix} \begin{bmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{bmatrix}, \tag{2}$$

where $t = 1, 2, \ldots, T$ stands for discrete time instants, $x_j(t)$, $1 \le j \le M$, is the jth mixed signal at time instant t, a_{ji} is the (j, i)th element in mixing matrix $\mathbf{A}$, and $s_i(t)$, $1 \le i \le N$, is the ith source signal at time instant t. We carry out separation frame by frame with the window length l, usually $l \ll T$, and adjacent frames are overlapped.

Let us define some notations as follows: $\Lambda_{ji} = \mathrm{diag}(a_{ji}, \ldots, a_{ji})$ denotes $l \times l$ matrix, where diag{} denotes a diagonal matrix, and

$$\mathbf{M} = \begin{bmatrix} \Lambda_{11} & \cdots & \Lambda_{1N} \\ \vdots & \vdots & \vdots \\ \Lambda_{M1} & \cdots & \Lambda_{MN} \end{bmatrix}. \tag{3}$$

We also define every frame of the mixed and source signal as column vectors.

$$\begin{aligned} \mathbf{b} &= \left[\mathbf{b}_1^T, \ldots, \mathbf{b}_M^T\right]^T, \\ \mathbf{f} &= \left[\mathbf{f}_1^T, \ldots, \mathbf{f}_N^T\right]^T, \end{aligned} \tag{4}$$

where $\mathbf{b}_j = [x_j(t), \ldots, x_j(t + l - 1)]^T$, $j = 1, \ldots, M$, denotes a frame of the jth mixed signal and $\mathbf{f}_i = [s_i(t), \ldots, s_i(t + l - 1)]^T$, $i = 1, \ldots, N$, denotes a frame of the ith source signal.

For every frame, (4) can be converted into the form

$$\mathbf{b} = \mathbf{M}\mathbf{f}. \tag{5}$$

We assume that the source $\mathbf{f}_i$ has a sparse representation on some dictionary $\mathbf{D}_i$

$$\mathbf{f}_i = \mathbf{D}_i\mathbf{g}_i, \tag{6}$$

where $\mathbf{g}_i$ is the sparse coefficient vector and $\mathbf{D}_i$ is the dictionary on which $\mathbf{f}_i$ has a sparse representation. Then $\mathbf{f}$ can be sparsely represented by

$$\mathbf{f} = \mathbf{D}\mathbf{g}, \tag{7}$$

where

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_i & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{D}_N \end{bmatrix} \tag{8}$$

is a dictionary $\mathbf{D}$ composed of $\mathbf{D}_i$ and

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}_1(t) \\ \vdots \\ \mathbf{g}_N(t) \end{bmatrix}. \tag{9}$$

Then

$$\mathbf{b} = \mathbf{M}\mathbf{f} = \mathbf{M}\mathbf{D}\mathbf{g}, \tag{10}$$

where $\mathbf{g}$ can be recovered by measurements $\mathbf{b}$ using an optimization process

$$\begin{aligned} &\min \quad \|\mathbf{g}\|_0 \\ &\text{s.t.} \quad \mathbf{b} = \mathbf{M}\mathbf{D}\mathbf{g} \end{aligned} \tag{11}$$

and $\|\cdot\|_0$ denotes the l_0-norm. For a general CS problem, obtaining the sparsest solution to an underdetermined system (11) is known to be a NP-hard problem, which requires intractable computations [1]. This has led to considerable efforts in developing tractable approximations to find the sparse solutions. In general, most of the sparse recovery algorithms can be categorized into one of the following three categories.

(i) The first one is generally known as greedy algorithms. These algorithms approximate the signals' support and amplitude iteratively. Orthogonal matching pursuit (OMP) [17] is a classical representative in this category.

(ii) The second category is associated with ℓ_1 regularized optimization method, which can be considered as the tightest convex relaxation of ℓ_0 norm. The basis pursuit (BP) [18] and basis pursuit denoising (BPDN) [19] are the classical ℓ_1 regularized optimization methods to recover sparse signals in noiseless and noisy environments, respectively.

(iii) The third category is based on the sparse Bayesian methodology. The problem is formulated as learning and inference in a probabilistic model. By properly choosing the hierarchical prior for the signals, sparsity can be imposed statistically [14, 20]. Sparse Bayesian learning is a classical method to recover the sparse signals by formulating a scaled Gaussian mixtures model [14]. The main advantages of the sparse Bayesian methods are their desirable statistical characteristics and flexibility in imposing prior information.

To solve $\mathbf{g}$ from (11), the observation $\mathbf{b}$, mixing matrix $\mathbf{M}$, and dictionary $\mathbf{D}$ are required, respectively. Many methods for dictionary training and estimating the mixing matrix have been reported [7, 21, 22]. For convenience and without losing generality, we utilize the K-SVD [23] dictionary composition and K-means unmixing estimation technique [12] that is to be described in Section 3. Then the method to solve $\mathbf{g}$ is described in Section 4.

The detailed procedures are summarized as follows.

Algorithm 1 (procedure for dictionary training and mixing matrix estimation).

(1) Every speaker's speech sample is taken as training data using K-SVD method. $\mathbf{D}_1, \ldots, \mathbf{D}_N$ are obtained.

(2) Mixing matrix is estimated in frequency domain by K-means unmixing estimation technique. The estimate for $\mathbf{A}$, that is, $\widehat{\mathbf{A}}$, is obtained.

(3) Format $\mathbf{D}_1, \ldots, \mathbf{D}_N$ and $\widehat{\mathbf{A}}$ into $\mathbf{D}$ and $\widehat{\mathbf{M}}$ according to the dimension of $\mathbf{A}$ and the selected frame window size.

(4) $\mathbf{b}$ is the mixed speech frame.

(5) The separated speech signal $\mathbf{g}$ can be recovered by solving (10) subject to (11) in a frame by frame manner.

3. Estimation of the Mixing Matrix

In TF domain, the mixing model in (1) can be written as

$$\mathbf{X} = \mathbf{AS}, \tag{12}$$

where $\mathbf{X}$ and $\mathbf{S}$ contain the STFT coefficients of $x(t)$ and $s(t)$, respectively. At every TF point (ω, t), we have

$$\begin{bmatrix} x_1(\omega,t) \\ \vdots \\ x_M(\omega,t) \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \vdots & \vdots \\ a_{M1} & \cdots & a_{MN} \end{bmatrix} \begin{bmatrix} s_1(\omega,t) \\ \vdots \\ s_N(\omega,t) \end{bmatrix}, \tag{13}$$

where a_{ji} is the (j,i)th element in mixing matrix $\mathbf{A}$ and they can be complex numbers as well. Denote $n = 1, \ldots, N$; then $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_n, \ldots, \mathbf{a}_N]$ is noninvertible. The sources are generally estimated under the assumption that the source signals are W-disjoint. Defining $\Omega_n(\omega)$ as the set of TF points in frequency bin ω where s_n is the dominant source, that is, $|s_n(\omega,t)| \gg |s_{n'}(\omega,t)|$ for $n' \neq n$, the mixing model in (13) can then be simplified as

$$\mathbf{x}(\omega,t) \approx \mathbf{a}_n s_n(\omega,t), \quad (\omega,t) \in \Omega_n(\omega). \tag{14}$$

The above equation implies that, given $\mathbf{x}(\omega,t)$, the vector $\mathbf{a}_n$ can be estimated up to an amplitude and phase ambiguity. Without loss of generality, this ambiguity is resolved by assuming that $\mathbf{a}_n$ is of unit norm with the first element being a positive and real value [10]. This can be achieved by normalizing the mixture sample vector as

$$\mathbf{x}(\omega,t) \longleftarrow \frac{\mathbf{x}(\omega,t)\, e^{-j\phi_{x_1(\omega,t)}}}{\|\mathbf{x}(\omega,t)\|}, \tag{15}$$

where $\phi_{x_1(\omega,t)}$ is the phase of the first entry of $\mathbf{x}(\omega,t)$ and $\|\cdot\|$ denotes the l_2-norm. The normalized $\mathbf{x}(\omega,t)$ can now be clustered into N clusters so that centroid of the nth cluster corresponds to the estimate of $\mathbf{a}_n$ [7, 10].

Conventional algorithms reported in [8–10] assume that the approximation in (14) holds for all the TF points. This, however, may not be true in a real environment. Instead of assuming that (14) applies for all the TF points, our proposed algorithm introduces a single source measure to quantify the validity of (14) for each TF point. Only TF points with a high value of confidence are used to estimate $\mathbf{a}_n$, based on which a more accurate mask can be computed to separate the sources.

3.1. The Proposed TF Points Selection. From (14), the corresponding $M \times M$ autocorrelation matrix can be expressed as

$$\begin{aligned} \mathbf{R}_{\mathbf{x}}(\omega,t) &= E\left\{\mathbf{x}(\omega,t)\,\mathbf{x}^H(\omega,t)\right\} \\ &= \mathbf{a}_n E\left\{s_q(\omega,t)\, s_q^*(\omega,t)\right\}\mathbf{a}_n^H \\ &= \mathbf{a}_n \mathbf{a}_n^H \sigma_q^2(\omega,t), \end{aligned} \tag{16}$$

where $E\{\cdot\}$ is the expectation operator, $\sigma_q^2(\omega,t) = E\{s_q(\omega,t)s_q^*(\omega,t)\}$, and $*$ is the conjugate operator. Therefore for single source points, we have

$$\operatorname{rank}\left(\mathbf{R}_{\mathbf{x}}(\omega,t)\right) = 1. \tag{17}$$

Considering that speech utterances are locally stationary [25],

$$\mathbf{R}_{\mathbf{x}}(\omega,t) \approx \sum_{\widetilde{t}=t-\Delta_t}^{t+\Delta_t} \mathbf{x}(\omega,\widetilde{t})\,\mathbf{x}^H(\omega,\widetilde{t}), \tag{18}$$

where $\Delta_t \geq 1$ specifies the number of neighboring TF points used to estimate $\mathbf{R}_{\mathbf{x}}(\omega,t)$ and is adjustable according to the time duration within which the source signals are considered to be stationary. It may not be proper to direct

use $\text{rank}(\mathbf{R}_{\mathbf{x}}(\omega,t)) = 1$ as the single source TF point measure because the energy of nondominant sources is not always zero. To deal with this issue, a modified TF point selection method is provided.

Assume that at a particular TF point (ω^o, t^o), source signals s_1 to s_n, $n \le N$, have nonzero energy and the signal s_i is assumed to be the dominant source which is γ dB higher than the other sources in terms of energy at (ω^o, t^o); that is,

$$\sigma_i^2(\omega^o, t^o) \ge 10^{\gamma/10} \sum \sigma_{\text{other}}^2(\omega^o, t^o). \tag{19}$$

Assuming the sources are uncorrelated, the autocorrelation matrix for this TF point can then be expressed as

$$\begin{aligned}\mathbf{R}_{\mathbf{x}}(\omega^o, t^o) &= \mathbf{a}_1\mathbf{a}_1^H\sigma_1^2(\omega^o, t^o) + \cdots + \mathbf{a}_i\mathbf{a}_i^H\sigma_2^2(\omega^o, t^o) \\ &\quad + \cdots + \mathbf{a}_N\mathbf{a}_N^H\sigma_n^2(\omega^o, t^o) \\ &= \mathbf{A}\boldsymbol{\Lambda}(\omega^o, t^o)\mathbf{A}^H,\end{aligned} \tag{20}$$

where

$$\begin{aligned}\mathbf{A} &= \begin{bmatrix}\mathbf{a}_1 & \cdots & \mathbf{a}_N\end{bmatrix}, \\ \boldsymbol{\Lambda}(\omega^o, t^o) &= \begin{bmatrix}\sigma_1^2(\omega^o, t^o) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_n^2(\omega^o, t^o)\end{bmatrix},\end{aligned} \tag{21}$$

and $\mathbf{a}_i = [a_{1i}, \ldots, a_{Mi}]^T$, $1 \le j \le N$, is the ith row of the mixing matrix $\mathbf{A}$. Note that the diagonal elements of the $n \times n$ matrix $\boldsymbol{\Lambda}(\omega^o, t^o)$ are nonzero. It is not proper to directly use (17) to determine single source point. Alternatively a continuous measure

$$\frac{\sigma_i^2(\omega^o, t^o)}{\sigma_{\text{others}}^2(\omega^o, t^o)} \tag{22}$$

is also not feasible since $\sigma_n^2(\omega^o, t^o)$ and $\mathbf{A}_n(\omega^o)$ are unknown in the speech separation problem.

Considering that the decomposition in (20) is similar to the singular value decomposition (SVD) of $\mathbf{R}_{\mathbf{x}}(\omega^o, t^o)$, the ratio of singular values of $\mathbf{R}_{\mathbf{x}}(K, \tau)$ is proposed as the single source TF point measure (SSTFM); that is,

$$\begin{aligned}\text{SSTFM}(\omega, t) &= \frac{\lambda_{\max}(\omega, t)}{\sum_{i=1}^{M}\lambda_i(\omega, t) - \lambda_{\max}(\omega, t)} \\ &= \frac{\lambda_{\max}(\omega, t)}{\text{trace}\left(\mathbf{R}_{\mathbf{x}}(\omega, t)\right) - \lambda_{\max}(\omega, t)},\end{aligned} \tag{23}$$

where $\lambda_i(\omega, t)$ is the ith singular value of $\mathbf{R}_{\mathbf{x}}(\omega, t)$ and $\lambda_{\max}(\omega, t)$ is the maximum singular value of $\mathbf{R}_{\mathbf{x}}(\omega, t)$. Application of SVD to detect single source points is valid since the separation problem at each single source TF point is an overdetermined problem. The TF points after selection are as illustrated in Figure 1.

Since the SSTFM provides a measure of $\mathbf{x}(\omega, t)$ being a single source point, only those $\mathbf{x}(\omega, t)$ with a high SSTFM should be used by the clustering algorithm to estimate $\mathbf{a}_n$. This can be achieved by selecting $\mathbf{x}(\omega, t)$ which has a SSTFM value above a threshold. This threshold can be predetermined or simply select the median value of $\text{SSTFM}(\omega, t)$ which will prevent cases of too few $\mathbf{x}(\omega, t)$ identified as single source point, which may degrade the performance of using clustering algorithm to estimate $\mathbf{a}_n$. As a result, for each frequency bin ω, the subset

$$\{\mathbf{x}(\omega, t) : \text{SSTFM}(\omega, t) \ge \text{threshold}\} \tag{24}$$

is used to estimate $\mathbf{a}_n$, that is, M in (11).

3.2. Estimating $\mathbf{a}_n$ of Mixing Matrix $\mathbf{A}$. After selecting the TF points, the next stage is the estimation of the mixing matrix. Here we are using the K-means clustering techniques [26]. It is noted that this may not be the best algorithm to cluster the samples as other algorithms can also be used. Since only the selected TF points are used in this step, the scatter plot has a clear orientation towards the directions of the column vectors in the mixing matrix. Hence these points are clustered into N groups. After clustering, the column vectors of the mixing matrix are determined by calculating the centroid of each cluster. Note that the points lying in the left-hand side of the vertical axis in the scatter diagram are mapped to the right-hand side (by changing their sign) before calculating the centroid.

3.3. Obtaining the Dictionary $\mathbf{D}$. We assume that before separating sources we have some speech samples as training data. In our approach, we directly train K-SVD dictionary on these speech samples using the source-trained strategy described in [27].

4. Speech Recovery

The matrix $\mathbf{MD}$ in (11) is not precisely known. The matrix obtained from the mixing matrix estimation step (Procedure (2) in Algorithm 1) contains errors due to time frequency point selection. From numerical experiments, recovering speech with the estimated matrix brings crosstalk residuals [11–13]. In theory, (11) with $\widehat{\mathbf{M}}$ is degenerated into

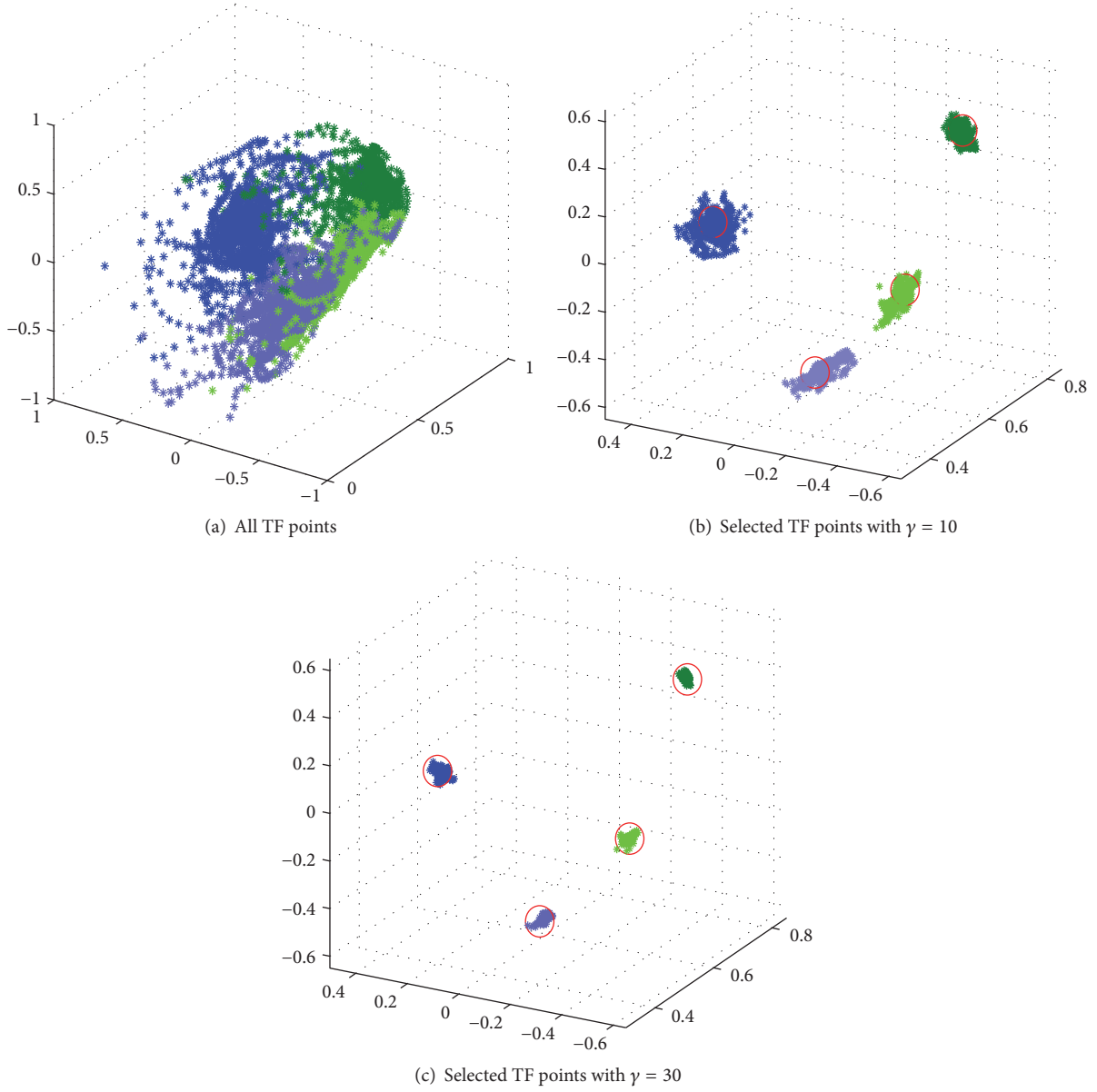
$$\begin{aligned}\min \quad & \|\mathbf{g}\|_0 \\ \text{s.t.} \quad & \mathbf{b} = \widehat{\mathbf{M}}\mathbf{D}\mathbf{g},\end{aligned} \tag{25}$$

where $\mathbf{g}$ solved from (25) is not the sparse solution with $\mathbf{MD}$ and observation $\mathbf{b}$. Instead, it is a sparse solution to a linear combination of trained dictionaries.

Thus to get the correct sparse solution, (10) should be alternatively expressed as

$$\mathbf{b} = \left(\widehat{\mathbf{M}} + \mathbf{E}_m\right)\mathbf{D}\mathbf{g}, \tag{26}$$

where $\mathbf{E}_m$ is $Ml \times Nl$ diagonal matrix to represent the difference between accurate $\mathbf{M}$ and estimated $\widehat{\mathbf{M}}$. In order to solve $\mathbf{g}$ from (26), we introduce a sparse Bayesian model as follows.



(a) All TF points

(b) Selected TF points with $\gamma = 10$

(c) Selected TF points with $\gamma = 30$

FIGURE 1: Three-dimensional view of TF points by plotting the real parts. Different clusters are marked with different colors and the red circles show the domain of ideal steering vectors of N sources.

4.1. Bayesian Model. Sparse Bayesian model [20] was derived from the research area of machine learning and has become a popular method for sparse signal recovery in CS. In this model, the sparse signal recovery problem is formulated from a Bayesian perspective while the sparsity information is exploited by assuming a sparse prior for the signal of interest. For instance, a Laplace prior [28] corresponds to the l_1 norm which has been widely studied in existing optimization approaches. Since exact Bayesian inference is typically intractable, approximation approaches to Bayesian inference have been adopted in [28]. One advantage of Bayesian CS compared to other CS methods is the flexibility of modeling sparse signals. It can promote the sparsity of its solution and exploit additionally known structures of the sparse signal [29] as in our case. Thus a sparse Bayesian model is selected to solve (26).

In our approach, we use the sparse Bayesian model in [28] to recover the sparse signals for sparser solutions. In other words, a sparer solution can be obtained by calibrating $\mathbf{E}_m$ in (26). The mixing matrix difference $\mathbf{E}_m$ is assumed to be an independent and identically distributed Gaussian noise and an empirical variance α_0. Probabilistic model is used for convenient inference. According to (26), the observation $\mathbf{b}$ is assumed to obey the following distribution:

$$p\left(\mathbf{b} \mid \mathbf{g}; \mathbf{E}_m\right) = \mathcal{N}\left(\mathbf{b} \mid \left(\widehat{\mathbf{M}} + \mathbf{E}_m\right)\mathbf{D}\mathbf{g}, \alpha_0 \mathbf{I}\right), \quad (27)$$

where α_0 is a noise variance between $\mathbf{b}$ and recovered signal and assumed to be known a prior [28].

Probabilistic model is used for the signal to impose the sparsity-inducing Laplace prior and enable convenient inference [28]. In the first stage of the signal model, the probability density function of the speech sources, $\mathbf{g}$, is given as

$$p(\mathbf{g} \mid \boldsymbol{\alpha}) = \prod_{i=1}^{L} \mathcal{N}(g_i \mid 0, \alpha_i), \tag{28}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_L)^T$ and L denotes the length of combined dictionary D. The hyperparameter, α_i for $i \in \{1 : L\}$, is modeled as an independent Gamma distribution

$$p(\boldsymbol{\alpha} \mid \lambda) = \prod_{i=1}^{L} \Gamma\left(\alpha_i \mid 1, \frac{\lambda}{2}\right). \tag{29}$$

Based on (28) and (29), the marginalized distribution $p(\mathbf{g} \mid \lambda)$ obeys a Laplace distribution [28]. Furthermore, the parameter, λ, controls the shape of the Laplace distribution and determines the sparsity of the signal $\mathbf{g}$. To conveniently learn λ, a Gamma distribution is assumed as

$$p(\lambda \mid \nu) = \Gamma\left(\lambda \mid \frac{\nu}{2}, \frac{\nu}{2}\right), \tag{30}$$

where ν is a parameter to be tuned and often set to be a small value as suggested in [28].

According to (28)–(30), the joint probability distribution conditioned on $\mathbf{E}_m$ can be derived as

$$\begin{aligned} &p(\mathbf{b}, \mathbf{g}, \boldsymbol{\alpha}, \lambda; \mathbf{E}_m) \\ &\quad = p(\mathbf{b} \mid \mathbf{g}; \mathbf{E}_m)\, p(\mathbf{g} \mid \boldsymbol{\alpha})\, p(\boldsymbol{\alpha} \mid \lambda)\, p(\lambda). \end{aligned} \tag{31}$$

4.2. Proposed Methods. An expectation maximization (EM) algorithm is implemented to solve (31). The EM algorithm requires the knowledge of the posterior distribution [30]

$$p(\mathbf{g}, \boldsymbol{\alpha}, \lambda \mid \mathbf{b}; \mathbf{E}_m) = \frac{p(\mathbf{g}, \boldsymbol{\alpha}, \lambda, \mathbf{b}; \mathbf{E}_m)}{p(\mathbf{b})}, \tag{32}$$

where $p(\mathbf{g}, \boldsymbol{\alpha}, \lambda, \mathbf{b}; \mathbf{E}_m)$ is obtained in (31) and $p(\mathbf{b}) = \iiint p(\mathbf{g}, \boldsymbol{\alpha}, \lambda, \mathbf{b}) d\mathbf{g}\, d\boldsymbol{\alpha}\, d\lambda$.

Subsequently, a distribution $q(\boldsymbol{\Theta})$, where $\boldsymbol{\Theta} = \{\mathbf{g}, \boldsymbol{\alpha}, \lambda\}$ denotes the hidden variables, is assumed to approximate the true posterior, which minimizes the Kullback-Leibler (KL) divergence between $q(\boldsymbol{\Theta})$ and the true posterior [31]:

$$\begin{aligned} q^*(\boldsymbol{\Theta}) &= \arg\min_{q(\boldsymbol{\Theta})} D_{\mathrm{KL}}(q(\boldsymbol{\Theta}) \mid p(\boldsymbol{\Theta} \mid \mathbf{b}; \mathbf{E}_m)) \\ &= \arg\min_{q(\boldsymbol{\Theta})} \left(\int q(\boldsymbol{\Theta}) \log \frac{q(\boldsymbol{\Theta})}{p(\boldsymbol{\Theta} \mid \mathbf{b}; \mathbf{E}_m)} d\boldsymbol{\Theta} \right). \end{aligned} \tag{33}$$

Then the hidden variable $\boldsymbol{\Theta}$ and parameter $\mathbf{E}_m$ can be iteratively updated by the following steps.

4.2.1. Expectation Stage. In this stage, the method assumes that $q(\boldsymbol{\Theta})$ has a factorization form:

$$q(\boldsymbol{\Theta}) = q(\mathbf{g}, \boldsymbol{\alpha}, \lambda) = q(\mathbf{g})\, q(\boldsymbol{\alpha})\, q(\lambda). \tag{34}$$

According to [31], the optimal distribution that minimizes (33) can be expressed as

$$\ln q^*(\Theta_K) = \langle \ln p(\mathbf{b}, \boldsymbol{\Theta}; \mathbf{E}_m) \rangle_{q(\boldsymbol{\Theta} \setminus \Theta_K)}, \tag{35}$$

where $\langle \cdot \rangle_{q(\boldsymbol{\Theta} \setminus \Theta_K)}$ denotes the expectation with respect to $q(\boldsymbol{\Theta} \setminus \Theta_K)$ and $\boldsymbol{\Theta} \setminus \Theta_K$ represents the set $\boldsymbol{\Theta}$ without Θ_K. By substituting $\mathbf{g}$, $\boldsymbol{\alpha}$, and λ into (35), respectively, the approximation is obtained from the following procedures.

(i) For $q^*(\mathbf{g})$, we have

$$\ln q^*(\mathbf{g}) = \langle \ln p(\mathbf{b} \mid \mathbf{g}; \mathbf{E}_m)\, p(\mathbf{g} \mid \boldsymbol{\alpha}) \rangle_{q(\boldsymbol{\alpha}) q(\lambda)} + c. \tag{36}$$

By substituting (27) and (28) into (36), $q^*(\mathbf{g})$ obeys the Gaussian distribution with a mean and a covariance matrix as

$$\begin{aligned} \boldsymbol{\mu} &= \frac{1}{\alpha_0 \boldsymbol{\Sigma} (\widehat{\mathbf{M}}\mathbf{D})^T \widehat{\mathbf{E}}_m \mathbf{b}}, \\ \boldsymbol{\Sigma} &= \left[\frac{1}{\alpha_0 (\widehat{\mathbf{M}}\mathbf{D})^T \widehat{\mathbf{E}}_m^2 (\widehat{\mathbf{M}}\mathbf{D})} + \left\langle \operatorname{diag}\left(\frac{1}{\boldsymbol{\alpha}}\right) \right\rangle_{q(\boldsymbol{\alpha})} \right]^{-1}. \end{aligned} \tag{37}$$

(ii) For $q^*(\boldsymbol{\alpha})$, we obtain

$$\ln q^*(\boldsymbol{\alpha}) = \langle \ln p(\mathbf{g} \mid \boldsymbol{\alpha})\, p(\boldsymbol{\alpha} \mid \lambda) \rangle_{q(\mathbf{g}) q(\lambda)} + c. \tag{38}$$

By substituting (28) and (29) into (38), α_n obeys a generalized inverse Gaussian distribution whose ith moment is expressed as [32]

$$\left\langle \alpha_n^i \right\rangle = \left(\frac{\langle g_n^2 \rangle_{q(g)}}{\langle \lambda \rangle_{q(\lambda)}} \right)^{i/2} \frac{\kappa_{0.5+i}\left(\sqrt{\langle \lambda \rangle_{q(\lambda)} \langle g_n^2 \rangle_{q(g)}} \right)}{\kappa_{0.5}\left(\sqrt{\langle \lambda \rangle_{q(\lambda)} \langle x_n^2 \rangle_{q(g)}} \right)}, \tag{39}$$

where κ_a is a Bessel function of the second kind.

(iii) For $q^*(\lambda)$, we have

$$\ln q^*(\lambda) = \langle \ln p(\boldsymbol{\alpha} \mid \lambda)\, p(\lambda; \nu) \rangle_{q(\mathbf{x}) q(\boldsymbol{\alpha})} + c. \tag{40}$$

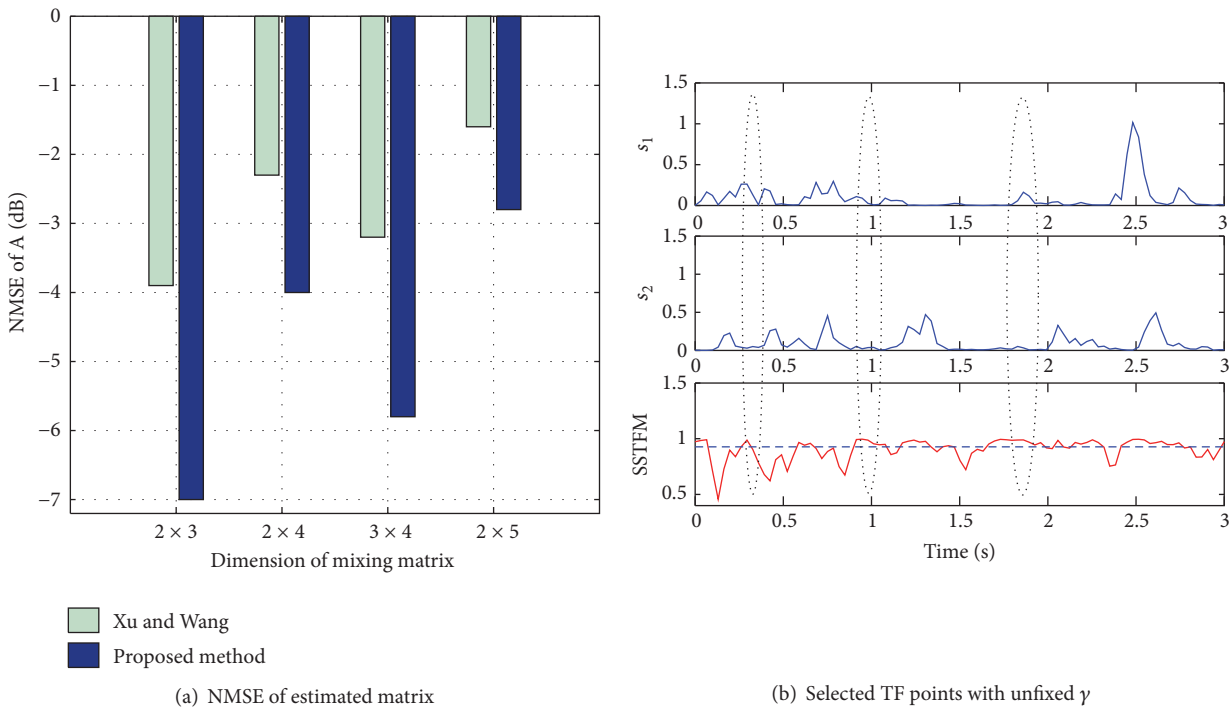
By substituting (29) and (30) into (40), it is shown that $q^*(\lambda)$ obeys a Gamma distribution with a mean

$$\langle \lambda \rangle = \frac{(2L + \nu)}{\left(\sum_{i=1}^{L} \langle \alpha_i \rangle_{q(\boldsymbol{\alpha})} + \nu \right)}. \tag{41}$$

The optimal approximated distribution $q^*(\boldsymbol{\Theta})$ can be obtained by iteratively calculating the above steps until convergence and each hidden variable is updated once before proceeding to the next stage.

4.2.2. Maximization Stage. According to [31], $\mathbf{E}_m$ is estimated by maximizing the expected log-likelihood as

$$\widehat{\mathbf{E}}_m = \arg\max_{\mathbf{E}_m} \langle \ln p(\mathbf{b}, \mathbf{g}, \boldsymbol{\alpha}, \lambda; \mathbf{E}_m) \rangle_{q(\mathbf{g}) q(\boldsymbol{\alpha}) q(\lambda)}. \tag{42}$$



(a) NMSE of estimated matrix

(b) Selected TF points with unfixed γ

Figure 2: NMSE comparison between [13] and proposed method and SSTFM by proposed method.

There exists a closed-form solution to (42) for updating $\mathbf{E}_m$; that is,

$$\widehat{\mathbf{E}}_m(i) = \frac{\left(\mathbf{b}_i - \left(\widehat{\mathbf{M}}\mathbf{D}\right)_i \mathbf{g}\right)\mathbf{D}_i\boldsymbol{\mu}}{\boldsymbol{\mu}^T\mathbf{D}_i^T\mathbf{D}_i\boldsymbol{\mu} + \operatorname{trace}\left(\mathbf{D}_i^T\mathbf{D}_i\boldsymbol{\Sigma}\right)}, \tag{43}$$

where $\mathbf{D}_i$ represents the ith row of matrix $\mathbf{D}$.

In summary, this proposed algorithm jointly estimates $\mathbf{g}$ and $\mathbf{E}_m$ to achieve sparsity and the procedures are listed in Algorithm 2.

Algorithm 2 (proposed method for solving $\mathbf{g}$).

(1) *Input*: $\mathbf{b}, \widehat{\mathbf{M}}\mathbf{D}, v$.

(2) *While* converge *do*

(3) Update $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ by (37).

(4) Update $\boldsymbol{\alpha}$ by (39)

(5) Update $\boldsymbol{\lambda}$ by (41)

(6) Update $\mathbf{E}_m$ by (43)

(7) *end while*

(8) *Output*: $\mathbf{g}$, $\mathbf{E}_m$

(9) The separated signals are recovered by $\mathbf{Dg}$.

5. Numerical Experiments

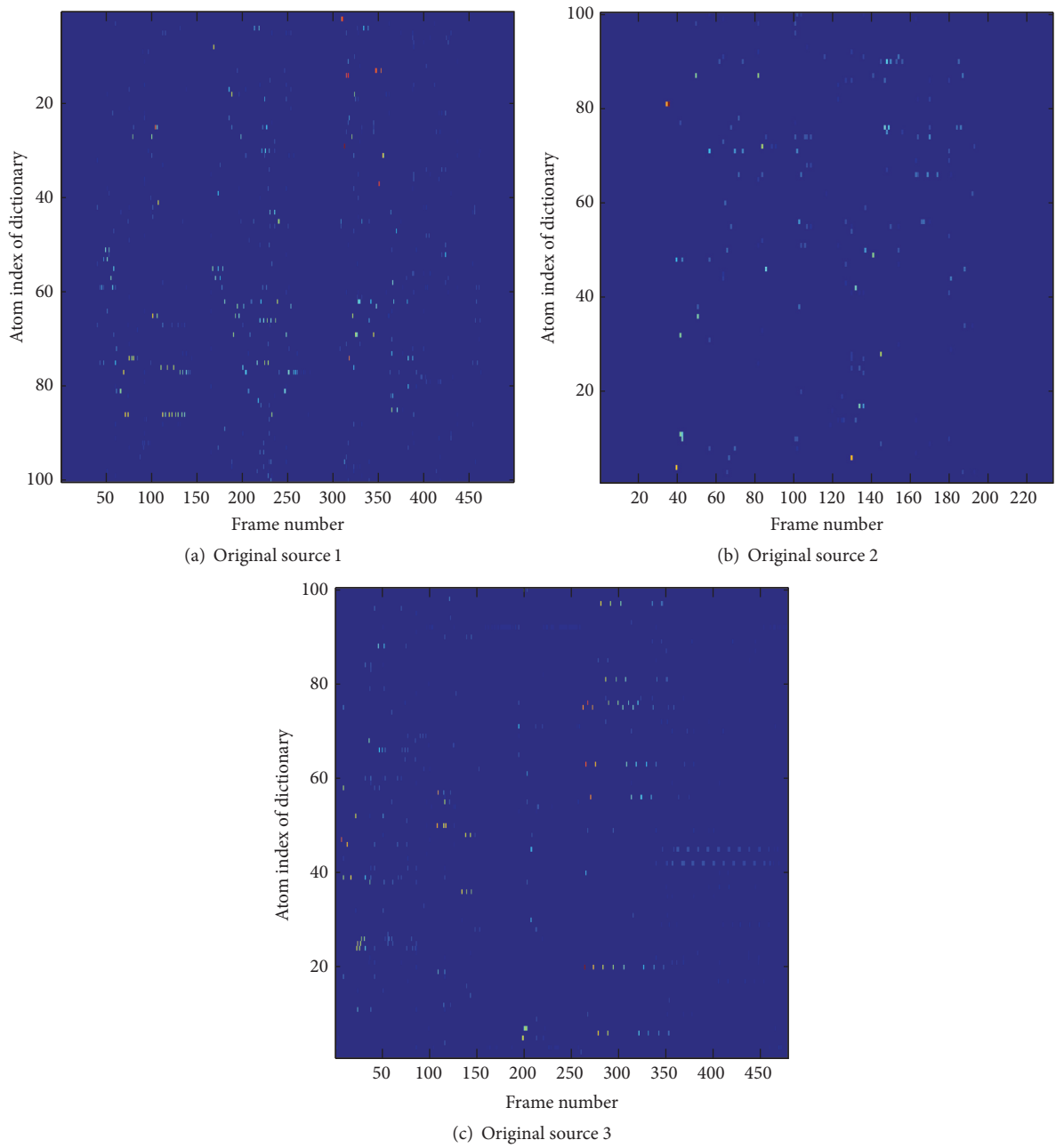
In this section we firstly describe the setting of our experiments and then comparisons are made in terms of the obtained performances by the proposed method and the other ones reported in the literature.

5.1. Estimating the Mixing Matrix. To compare our proposed modified unmixing method with algorithm described in [13], we use the mixing matrix randomly generated and mix the clean speeches from [33] to get the speech mixtures. The average normalized mean square error (NMSE) over 50 trials is presented in Figure 2(a) where the median value of SSTFM values is used as threshold rather than a predetermined one, and the value of Δ_t in (18) is 2. Figure 2(a) shows that the proposed TF selection method has a smaller NMSE with various dimensions of mixing matrix. However the exact mixing matrix is not obtained because the NMSEs obtained by both methods are always larger than zero, which motivates the model reported in (26). Figure 2(b) illustrates two source signals at frequency bin ω and the estimated SSTFM using the mixtures in a simulation trial. It can been seen that the proposed method is effective in identifying the single source points with unfixed SSTFM, as marked by the dashed ovals.

5.2. Speech Signal Recovery. The speech sample is downloaded from the database of the source separation evaluation campaign [33]. For every speaker, we have sentences for testing and training separately. A sparse dictionary is prepared for every speaker by using K-SVD Algorithm [23]. Figure 3 illustrates the sparse coefficients obtained from the source signals. The parameters for K-SVD are in Table 1.

The mixing matrix is randomly generated. Two examples of random 2×3 and 2×4 mixed matrices and the corresponding estimated matrices by modified method are shown as reference with permutation ambiguity ignored.

$$\mathbf{A}_{2\times 3} = \begin{bmatrix} 0.3420 & 0.6428 & 0.9239 \\ 0.9397 & 0.7660 & 0.3826 \end{bmatrix},$$



(a) Original source 1

(b) Original source 2

(c) Original source 3

Figure 3: The sparse coefficients of original sources solved from dictionary trained with K-SVD.

$$\widehat{\mathbf{A}}_{2\times3} = \begin{bmatrix} 0.3425 & 0.6422 & 0.9239 \\ 0.9395 & 0.7665 & 0.3826 \end{bmatrix},$$

$$\mathbf{A}_{2\times4} = \begin{bmatrix} 0.3620 & 0.6275 & 0.7896 & 0.9184 \\ 0.9332 & 0.7786 & 0.6136 & 0.3957 \end{bmatrix},$$

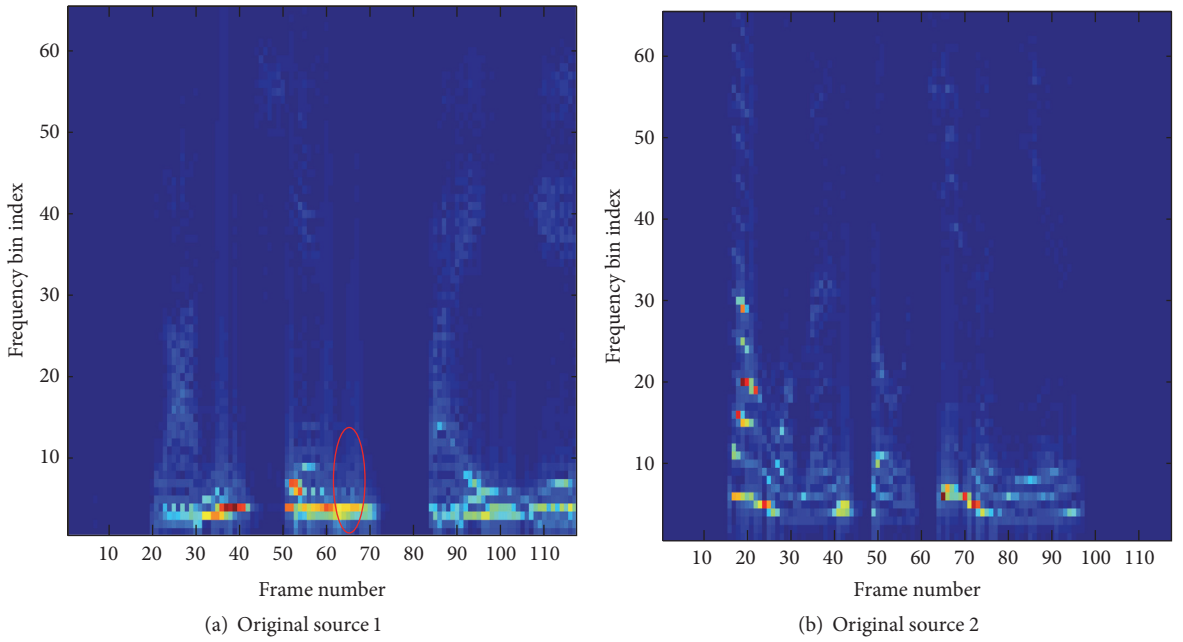
$$\widehat{\mathbf{A}}_{2\times4} = \begin{bmatrix} 0.3709 & 0.6274 & 0.7913 & 0.9045 \\ 0.9287 & 0.7787 & 0.6114 & 0.4265 \end{bmatrix}. \quad (44)$$

Figures 5 and 6 show the recovery signal in STFT domain using sparse Bayesian [14] and proposed method,

Table 1: The parameters for K-SVD training.

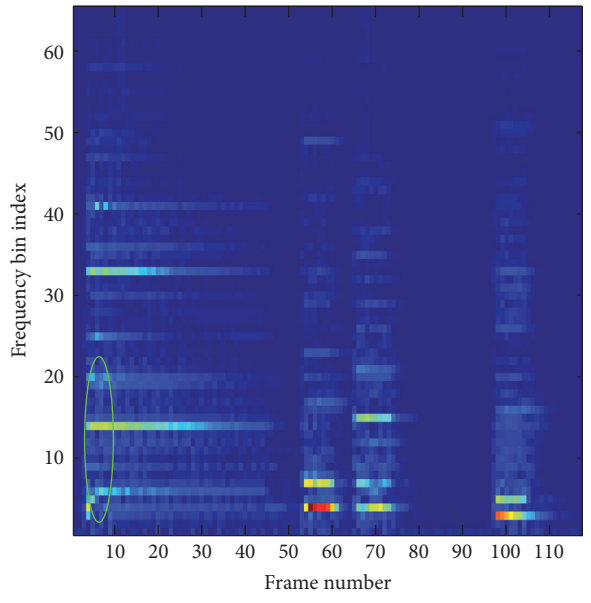
The number of training data	10 speeches for every speaker
Window length	1024
Dictionary size	$1024 * 3072$
The number of iterations	200
Window overlap	50%
Sampling rate	8000

respectively. Compared with original sources in Figure 4, the significant differences are marked by the highlighted ovals. In Figure 5(b), the interference inside the green ovals mainly comes from source 1 (also see Figure 4(a)) while the points



(a) Original source 1

(b) Original source 2



(c) Original source 3

Figure 4: Original speech signal in STFT domain.

in red ovals are mainly from source 3 (also see Figure 4(a)). With our proposed method, these interferences are avoided by calibrating the mixing matrix while solving the sparse solution.

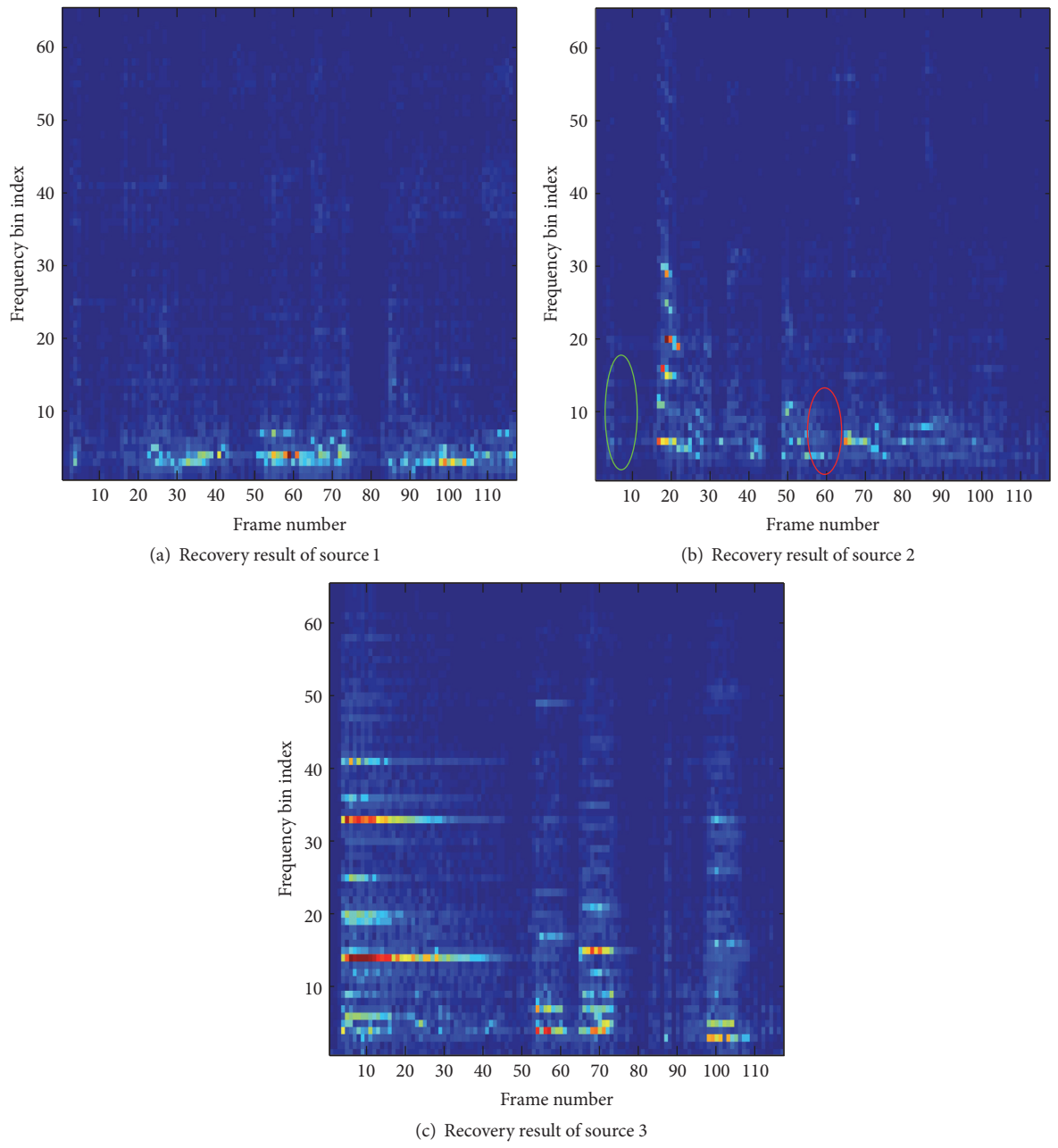
The separation performance is quantified in terms of source-to-interference ratio (SIR), source-to-distortion ratio (SDR), and source-to-artifacts ratio (SAR). Without loss of generality, we assume the separated output $\widehat{s}_q(t)$ corresponding to the source signal $s_q(t)$ for ease presentation. The three performance measures first decompose the qth source estimate $y_q(t)$ using orthogonal projections as [34]

$$\widehat{S}_q(t) = s_{\text{target}_q}(t) + e_{\text{interf}_q}(t) + e_{\text{artif}_q}(t) + e_{\text{noise}_q}(t), \quad (45)$$

where $s_{\text{target}_q}(t)$ is the portion attributing to $s_q(t)$, $e_{\text{interf}_q}(t)$ is the interference from other sources, $e_{\text{artif}_q}(t)$ is the artifacts introduced by the separation algorithm, and $e_{\text{noise}_q}(t)$ is the noise effect. The SIR, SDR, and SAR for source q are defined as

$$\text{SIR}_q = 10\log_{10}\frac{\sum_t s^2_{\text{target}_q}(t)}{\sum_t e^2_{\text{interf}_q}(t)},$$

$$\text{SDR}_q = 10\log_{10}\frac{\sum_t s^2_{\text{target}_q}(t)}{\sum_t \left(e_{\text{interf}_q}(t) + e_{\text{artif}_q}(t) + e_{\text{noise}_q}(t)\right)^2},$$



(a) Recovery result of source 1

(b) Recovery result of source 2

(c) Recovery result of source 3

Figure 5: Speech signal recovered by sparse Bayesian method in STFT domain.

$$\mathrm{SAR}_q = 10\log_{10}\frac{\sum_t \left(s_{\mathrm{target}_q}(t) + e_{\mathrm{interf}_q}(t) + e_{\mathrm{noise}_q}(t)\right)^2}{\sum_t e^2_{\mathrm{artif}_q}(t)}. \tag{46}$$

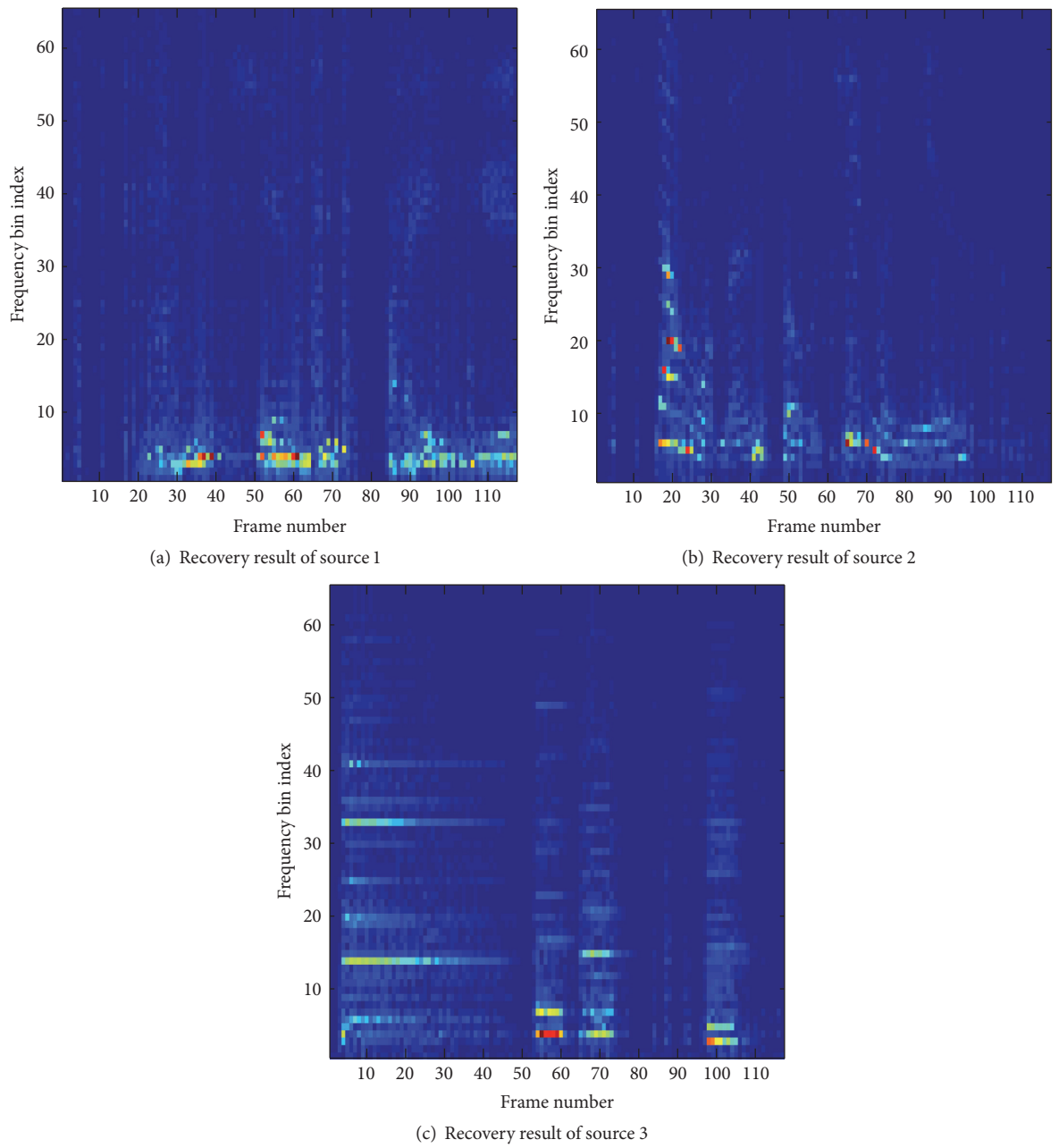
Since SDR considers both interference and artifacts, it is expected to be a more comprehensive criterion compared with SIR and SAR [34]. All the above measures can be computed using the *BSS-EVAL* Toolbox [35]. Note that in our noiseless simulations, $e_{\mathrm{noise}_q}(t) = 0$, which will not affect the three criteria defined above.

The performance gain of the proposed algorithm compared to other algorithms is illustrated in Table 2.

From Table 2, it is seen that in the underdetermined situation (mixing matrix = 2×3 and 2×4), the proposed method increases both SIR and SAR, which means that, by introducing autocalibrating $\mathbf{E}_m$ to (11) to obtain a sparser representation, both interferences and artifacts are effectively suppressed. The proposed algorithm achieves worse performance than that obtained by using a known mixing matrix, as shown in the last column of Table 2 because that $\mathbf{g}$ is solved by using $\mathbf{E}_m + \widehat{\mathbf{M}}\mathbf{D}$ not $\mathbf{M}\mathbf{D}$. Note that even using a known mixing matrix, the recovery signal still have interference, distortion, and artifacts for two reasons. Firstly for mixtures separation,

Table 2: Performance comparison of different algorithms.

	Basis pursuit [24]	Sparse Bayesian	Proposed method	With known mixing matrix
Mixing matrix = 2 × 3				
Average SAR of $\widehat{\mathbf{S}}$	9.6	9.7	**9.9**	10.8
Average SDR of $\widehat{\mathbf{S}}$	7.2	7.2	**7.3**	8.1
Average SIR of $\widehat{\mathbf{S}}$	13.4	13.4	**13.8**	14.5
Mixing matrix = 2 × 4				
Average SAR of $\widehat{\mathbf{S}}$	6.9	6.9	**7.0**	7.7
Average SDR of $\widehat{\mathbf{S}}$	5.2	5.2	**5.2**	6.1
Average SIR of $\widehat{\mathbf{S}}$	11.8	11.8	**12.1**	13.4



(a) Recovery result of source 1

(b) Recovery result of source 2

(c) Recovery result of source 3

Figure 6: Speech signal recovered by proposed method in STFT domain.

g is solved by using **MD** rather than the trained dictionary **D** directly. Secondly, the speech sources used for dictionary training are different from the those used for separation evaluation.

6. Conclusions

We have presented the compressed sensing based algorithm to solve the problem of instantaneous underdetermined speech separation. Since exact mixing matrix is unreachable, the sparse Bayesian learning model is used and the separated speeches are recovered from approximated posterior distribution derived with the EM method. The proposed one operates in a statistical manner to achieve a sparser estimation. Numerical experimental results show that the proposed algorithm provides better estimation performance than the other methods reported in the literature.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This research work is supported by the National Natural Science Foundation of China (no. 61571174) and Zhejiang Provincial Natural Science Foundation of China (no. LY15F010010).

References

[1] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[2] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[3] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007.

[4] P. Comon and C. Jutten, Eds., *Handbook of Blind Source Separation: Independentcomponent Analysis and Applications*, Academic Press, 2010.

[5] M. S. Pedersen, J. Larsen, U. Kjems, and L. C. Parra, *A Survey of Convolutive Blind Source Separation Methods*, Springer, New York, NY, USA, 2007.

[6] A. Hyvrinen, K. Juhu, and O. Erkki, *Independent Component Analysis*, Wiley-Interscience, 2001.

[7] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[8] S. Araki, H. Sawada, R. Mukai, and S. Makino, "Underdetermined blind sparse source separation for arbitrarily arranged multiple sensors," *Signal Processing*, vol. 87, no. 8, pp. 1833–1847, 2007.

[9] V. G. Reju, S. N. Koh, and I. Y. Soon, "Underdetermined convolutive blind source separation via time-frequency masking," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 1, pp. 101–116, 2010.

[10] H. Sawada, S. Araki, and S. Makino, "Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 3, pp. 516–527, 2011.

[11] G. Bao, Z. Ye, X. Xu, and Y. Zhou, "A compressed sensing approach to blind separation of speech mixture based on a two-layer sparsity model," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 21, no. 5, pp. 899–906, 2013.

[12] T. Xu and W. Wang, "A compressed sensing approach for underdetermined blind audio source separation with sparse representation," in *Proceedings of the IEEE/SP 15th Workshop on Statistical Signal Processing (SSP '09)*, pp. 493–496, Cardiff, UK, September 2009.

[13] T. Xu and W. Wang, "A block-based compressed sensing method for underdetermined blind speech separation incorporating binary mask," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '10)*, pp. 2022–2025, Dallas, Tex, USA, March 2010.

[14] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.

[15] L. Zhao, L. Wang, G. Bi, L. Zhang, and H. Zhang, "Robust frequency-hopping spectrum estimation based on sparse bayesian method," *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 781–793, 2015.

[16] L. Zhao, L. Wang, G. Bi, S. Li, L. Yang, and H. Zhang, "Structured sparsity-driven autofocus algorithm for high-resolution radar imagery," *Signal Processing*, vol. 125, pp. 376–388, 2016.

[17] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[18] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[19] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[20] M. Tipping, "Sparse bayesian learning and the relevance vectormachine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[21] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: learning sparse dictionaries for sparse signal approximation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1553–1564, 2010.

[22] M. G. Jafari and M. D. Plumbley, "Fast dictionary learning for sparse representations of speech signals," *IEEE Journal on Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 1025–1031, 2011.

[23] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[24] E. van den Berg and M. P. Friendlander, "Probing the pareto frontier for basis pursuit solutions," UBC Computer Science Tech. Rep TR-2008-01, 2008.

[25] E. Vincent, S. Arberet, and R. Gribonval, "Underdetermined instantaneous audio source separation via local gaussian modeling," in *Independent Component Analysis and Signal Separation*, T. Adali, C. Jutten, J. M. T. Romano, and A. K. Barros, Eds., vol. 5441 of *Lecture Notes in Computer Science*, pp. 775–782, Springer, New York, NY, USA, 2009.

[26] R. Xu and D. Wunsch II, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.

[27] T. Xu and W. Wang, "Methods for learning adaptive dictionary in underdetermined speech separation," in *Proceedings of the 21st IEEE International Workshop on Machine Learning for Signal Processing (MLSP '11)*, pp. 1–6, September 2011.

[28] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[29] L. He and L. Carin, "Exploiting structure in wavelet-based Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 57, no. 9, pp. 3488–3497, 2009.

[30] L. Zhao, G. Bi, L. Wang, and H. Zhang, "An improved auto-calibration algorithm based on sparse bayesian learning framework," *IEEE Signal Processing Letters*, vol. 20, no. 9, pp. 889–892, 2013.

[31] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference: Life after the EM algorithm," *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, 2008.

[32] B. Jørgensen, *Statistical Properties of the Generalized Inverse Gaussian distribution*, Springer, New York, NY, USA, 1982.

[33] S. Araki, F. Nesta, E. Vincent et al., "The 2011 signal separation evaluation campaign (sisec2011): -audio source separation," in *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation ((LVA/ICA '12)*, pp. 414–422, Springer, Tel Aviv, Israel, March 2012.

[34] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[35] C. Fevotte, R. Gribonval, and E. Vincent, "BSS EVAL toolbox user guide," Tech. Rep. 1706, IRISA, Rennes, France, 2005.