A Compressed Sensing Approach to Blind Separation of Speech Mixture Based on a Two-Layer Sparsity Model

Guangzhao Bao, *Student Member, IEEE*, Zhongfu Ye, Xu Xu, and Yingyue Zhou

***Abstract*—This paper discusses underdetermined blind source separation (BSS) using a compressed sensing (CS) approach, which contains two stages. In the first stage we exploit a modified K-means method to estimate the unknown mixing matrix. The second stage is to separate the sources from the mixed signals using the estimated mixing matrix from the first stage. In the second stage a two-layer sparsity model is used. The two-layer sparsity model assumes that the low frequency components of speech signals are sparse on K-SVD dictionary and the high frequency components are sparse on discrete cosine transformation (DCT) dictionary. This model, taking advantage of two dictionaries, can produce effective separation performance even if the sources are not sparse in time-frequency (TF) domain.**

***Index Terms*—Compressed sensing, K-means, K-SVD, monochannel dictionary, multichannel dictionary, two-layer sparsity model, underdetermined blind source separation.**

I. Introduction

THE task of blind source separation (BSS) is to recover the sources using the observable signals. The noise-free instantaneous mixing model of BSS 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 BSS, i.e., $M < N$. Considering the difficulty of this problem, we only study the case where $M = 2$ and $N = 3$.

Many underdetermined BSS algorithms have been developed for speech separation. To the best of my knowledge, most of the proposed methods rely on the sparsity of signals in some domain, such as the time-frequency (TF) domain [1], [2]. Some authors also attempted to use trained dictionaries [3], [4] to replace discrete cosine transformation (DCT) or fast Fourier transformation (FFT) dictionary which is fixed, and showed that the trained dictionaries perform better than the fixed dictionaries. This is due to their full adaptability; but the adaptability usually results in high computational expense.

Degenerate unmixing estimation technique (DUET) by A. Jourjine *et al.* [1], one of the well-known underdetermined BSS algorithms, assumes that speech sources meet the W-Disjoint orthogonality (W-DO) assumption, i.e., the TF representations of the sources do not overlap. Under this assumption, the amplitude and phase parameters are estimated by creating the amplitude-phase histogram from two mixtures, in which every peak corresponds to a source, and binary TF masking is used to separate the sources. In reality the W-DO assumption is too strict, and sources usually overlap in the TF domain to a certain extent. Ö. Yılmaz *et al.* [5] introduced the concept of approximate W-DO and proposed a modified version of DUET. There are also some other improved variants of DUET. J. Han *et al.* [6] refined the mixing matrix by taking advantage of the harmonic structure of the harmonic sources and improve the results by an iterative way. Y. Lv *et al.* [7] relaxed the assumption by allowing the sources to overlap in some TF regions and proposed an explicit treatment of the overlapped TF points, leading to significant signal to interference ratio (SIR) improvements.

In recent years, compressed sensing (CS) theory has attracted a great deal of attention [8]–[11]. It provides potentially a powerful framework for computing a sparse representation of signals. In the underdetermined BSS problem, the underdetermined mixture is a form of compressed sampling, and therefore CS theory can be utilized to solve the problem. T. Blumensath *et al.* pointed out several similarities between CS and source separation [21]. T. Xu *et al.* developed a framework for this problem based on CS using fixed dictionary [12], while they used trained dictionary instead in [13] and proposed a multi-stage method for underdetermined BSS using block-based CS incorporating binary mask in [14].

Algorithms about designing an appropriate dictionary have been developed recently. M. Aharon *et al.* proposed the K-SVD algorithm [15]. On this basis, R. Rubinstein *et al.* improved K-SVD by approximate K-SVD [16], which is quite efficient. Later the same authors proposed a modified version of K-SVD, namely "double sparsity" dictionary [17]. Some other dictionary learning methods also have been developed such as the principal component analysis (PCA) dictionary [18], [19] and the greedy adaptive dictionary (GAD) [20].

Manuscript received January 13, 2012; revised May 23, 2012, September 02, 2012, and November 09, 2012; accepted November 28, 2012. Date of publication December 20, 2012; date of current version February 01, 2013. This work was supported by the Science and Technology Plan Project of Anhui Province of China (No. 11010202191). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jingdong Chen.

The authors are with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China, and also with National Engineering Laboratory for Speech and Language Information Processing, Hefei 230027, China (e-mail: gzbao@mail.ustc.edu.cn; yezf@ustc.edu.cn; xxu@ustc.edu.cn; zyyzhou@mail.ustc.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2012.2234110

In this paper, we present a two-stage approach, which first estimates the mixing matrix and then recovers the underlying sources. In the recovering stage, we assume that the sources satisfy a two-layer sparsity model: the low frequency components are sparse on K-SVD dictionary and the high frequency components are sparse on DCT dictionary. This model, taking advantage of two dictionaries, can improve the performance.

This paper is organized as follows: in Section II, we introduce the CS framework of BSS and explain the meaning of sparsity in BSS. In Section III, we give a concise method to estimate the mixing matrix and take advantage of the two-layer sparsity model to recover the underlying sources in view of CS theory. In Section IV, we compare the performance of our approach with some other recently proposed methods. Conclusions are drawn in Section V.

II. CS Framework and Sparsity

In this section, we begin our description by introducing the CS framework of BSS. Then we explain the meaning of sparsity in BSS which can give the motivation of our proposed sparsity model.

A. The CS Framework of BSS

Considering a noise-free model with $M = 2$ and $N = 3$, we can expand (1) as follows:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix}, \tag{2}$$

where $t = 1, 2, \ldots, T$ stand for discrete time instants, $x_j(t)$ is the jth mixed signal at time instant t, $s_i(t)$ 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 an overlap.

We define some notations here: $\Lambda_{ji} = \mathrm{diag}(a_{ji}, \ldots, a_{ji})$ denotes an $l \times l$ matrix, where $\mathrm{diag}\{\bullet\}$ denotes a diagonal matrix, and $\mathbf{M} = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} & \Lambda_{13} \\ \Lambda_{21} & \Lambda_{22} & \Lambda_{23} \end{bmatrix}$. We also define every frame of the mixed (or source) signal as a column vector as follows.

$$\mathbf{b} = \left[\mathbf{b}_1^T, \mathbf{b}_2^T\right]^T, \mathbf{f} = \left[\mathbf{f}_1^T, \mathbf{f}_2^T, \mathbf{f}_3^T\right]^T, \tag{3}$$

where $\mathbf{b}_j = [x_j(t), \ldots, x_j(t+l-1)]^T$, $j = 1,2$ denotes a frame of the jth mixed signal and $\mathbf{f}_i = [s_i(t), \ldots, s_i(t+l-1)]^T$, $i = 1,2,3$ denotes a frame of the ith source signal.

For every frame, (2) can be converted to the following form:

$$\mathbf{b} = \mathbf{M}\mathbf{f}. \tag{4}$$

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

$$\mathbf{f}_i = \mathbf{D}_i \mathbf{g}_i, \tag{5}$$

where $\mathbf{g}_i$ is the sparse coefficient vector, $\mathbf{D}_i$ is the dictionary on which $\mathbf{f}_i$ has a sparse representation. $\mathbf{D}_i$ can be either identical or not for different i, depending on different strategies. Then $\mathbf{f}$ can be sparsely represented on a dictionary $\mathbf{D}$ which is composed of $\mathbf{D}_i$ as follows:

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

where $\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{D}_3 \end{bmatrix}$, $\mathbf{g} = \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \mathbf{g}_3 \end{bmatrix}$.

Then

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

where $\widehat{\mathbf{M}} = \mathbf{M}\mathbf{D}$. One can clearly observe that this equation is similar to the model of CS in which $\mathbf{b}$ is the compressed vector of signal $\mathbf{f}$ and $\widehat{\mathbf{M}}$ is the CS matrix [11].

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

$$\min \|\mathbf{g}\|_0 \quad s.t. \quad \mathbf{b} = \widehat{\mathbf{M}}\mathbf{g}, \tag{8}$$

where $\|\cdot\|_0$ denotes the ℓ_0-norm. The solution to the above optimization is an NP-hard problem. However, several papers [11], [12], [21] have shown that the ℓ_0 minimization problem can be converted to solving the following ℓ_1 minimization problem:

$$\min \|\mathbf{g}\|_1 \quad s.t. \quad \mathbf{b} = \widehat{\mathbf{M}}\mathbf{g}, \tag{9}$$

where $\|\cdot\|_1$ denotes the ℓ_1-norm. It is convex so that we can use the basis pursuit (BP) method [22], [23] to solve the problem and find the sparsest representation of $\mathbf{f}$. It should be noted that other methods like orthogonal matching pursuit (OMP) [24], [25], matching pursuit (MP) [26], focal underdetermined system solver (FOCUSS) [27] and stagewise orthogonal matching pursuit (StOMP) [28] can also be used to solve this problem.

According to the CS theory, if $\widehat{\mathbf{M}}$ obeys a uniform uncertainty principle [8], [9], [11], [12] and also $\mathbf{g}$ is sparse [8], [9], [11], [29], $\mathbf{g}$ can be successfully restored. It means that every submatrix of $\widehat{\mathbf{M}}$ has to be well designed to obey a Restricted Isometry Property (RIP) [11], [12].

B. Sparsity

As shown previously, successful recovery requires the coefficient vector $\mathbf{g}$ to be sparse. The sparsity in the CS theory can be measured by several measurements [29]. But in the situation of BSS, sparsity has a new meaning: different sources contain different components corresponding to the atoms of dictionary. We list these two meanings as follows:

1) In all the elements of $\mathbf{g}$, the nonzero, more precisely, weighty elements only occupy a small fraction of all the elements.
2) Three sources have different distributions on the dictionary, or in other words, different sources occupy different dictionary bands.

The first meaning of sparsity is understandable, according to the CS theory, while the second meaning is essentially the same as W-DO [1], [5]–[7] if the dictionary is chosen to be frequency

dictionary (like FFT dictionary). If different sources occupy different frequency bands, we say they are sparse. The more different frequency bands they occupy, the sparser they become. But in this paper we do not only use fixed dictionary like FFT and DCT dictionary, but also use the trained dictionary.

III. Two Stages Approach to Separate Sources

In this section, we introduce an improved method of estimating the mixing matrix. Inspired by the meanings of sparsity, we introduce the dictionary based on a two-layer sparsity model.

A. Estimating the Mixing Matrix by Modified K-Means

The mixing matrix is usually previously unknown. In this stage, we estimate the mixing matrix by using singular value decomposition (SVD) and K-means clustering algorithm on the short-time Fourier transform (STFT) coefficients. Some authors have shown that speech signals are sparser in the TF domain than in the time domain [1], [12]. In the TF domain,

$$\tilde{\mathbf{X}} = \mathbf{A}\tilde{\mathbf{S}}, \tag{10}$$

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

$$\begin{bmatrix} \tilde{X}_1(\omega,t) \\ \tilde{X}_2(\omega,t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{S}}_1(\omega,t) \\ \tilde{\mathbf{S}}_2(\omega,t) \\ \tilde{\mathbf{S}}_3(\omega,t) \end{bmatrix}. \tag{11}$$

Some researchers assumed that at a large proportion of TF points (ω, t), only one source is active, namely, it is sparse, so that $\tilde{X}_1(\omega,t)/\tilde{X}_2(\omega,t)$ is approximately equivalent to a_{1i}/a_{2i}, $i = 1, 2, 3$. Hence if $a_{1i}/a_{2i} \neq a_{1i'}/a_{2i'}$, $i \neq i'$, $i, i' = 1, 2, 3$, a scatter plot of $\tilde{X}_1$ vs $\tilde{X}_2$ would cluster into three distinct lines such that the ith source corresponds to the line with gradient a_{1i}/a_{2i}, $i = 1, 2, 3$. Then one can use the K-means algorithm [12] on $\tilde{X}_1(\omega,t)/\tilde{X}_2(\omega,t)$ to obtain the three clusters and estimate every column of mixing matrix with an amplitude uncertainty.

Unfortunately, the assumed sparsity is hard to satisfy for all points. In [5], the authors estimated the mixing matrix by constructing a high resolution histogram and then smoothing it. In [6], the mixing matrix was refined by taking advantage of the harmonic structure of the harmonic sources. In this work we exploit SVD instead of histogram: we firstly delete the insufficiently sparse TF points and use the remaining points for clustering (see Fig. 1). We define the convariance matrix of TF mixture vectors:

$$\mathbf{R}_{\tilde{\mathbf{X}}} = E[\tilde{\mathbf{X}}\tilde{X}^H] = \mathbf{A}\mathbf{R}_{\tilde{\mathbf{S}}}\mathbf{A}^H, \tag{12}$$

where $\tilde{\mathbf{X}} = \begin{bmatrix} \tilde{X}_1(\omega,t) \\ \tilde{X}_2(\omega,t) \end{bmatrix}$ and $\tilde{X}_j(\omega,t)$, assumed to be zero mean, is the STFT of jth mixture signal at point (ω, t).We exploit SVD to $\mathbf{R}_{\tilde{\mathbf{X}}}$:

$$\mathbf{R}_{\tilde{\mathbf{X}}} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^H, \tag{13}$$

where $\mathbf{U} = [\mathbf{u_1}, \mathbf{u_2}]$ is the eigenvector matrix, and $\mathbf{\Sigma} = \mathrm{diag}\{\sigma_1^2, \sigma_2^2\}, \sigma_1^2 \geq \sigma_2^2$ is the eigenvalue matrix. Here we estimate $\mathbf{R}_{\tilde{\mathbf{X}}}$ by frequency average in a short time (ST) window, i.e., $\mathrm{E}\{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H\} \approx (1/l)\sum_\omega \tilde{\mathbf{X}}(\omega,t)\tilde{\mathbf{X}}^H(\omega,t)$, where l is the length of an ST window.

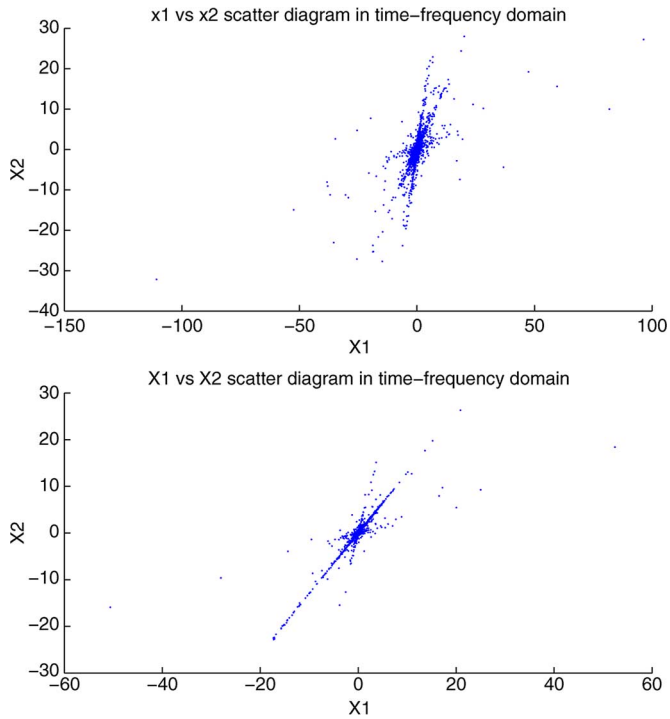


Fig. 1. Clustering in [12] uses all the TF points to estimate the mixing matrix (upper); our proposed clustering uses the sparse TF points to estimate the mixing matrix (lower).

If in some ST window only the ith source is active, then

$$s_i^2 \neq 0, s_{i'}^2 = 0, (i' \neq i), \tag{14}$$

$$\mathbf{R}_{\tilde{\mathbf{X}}} = \mathbf{a}_i s_i^2 \mathbf{a}_i^H \tag{15}$$

where s_i^2 is the power of ith source. Under this assumption, the rank of $\mathbf{R}_{\tilde{\mathbf{X}}}$ is 1 obviously and $\sigma_1 > 0, \sigma_2 = 0$. Given this, (12) can be simplified as follows:

$$\mathbf{R}_{\tilde{\mathbf{X}}} = \sigma_1^2 \mathbf{u}_1 \mathbf{u}_1^H. \tag{16}$$

Comparing (15) with (16), one can see that $\mathbf{u}_1$ is an estimate of $\mathbf{a}_i$. We find that every ST window corresponds to a $\mathbf{u}_1$ and all of the $\mathbf{u}_1$ cluster into three different vectors corresponding to the columns of mixing matrix.

As mentioned previously, sparsity assumption is not always satisfied. If we use all the $\mathbf{u}_1$ as cluster samples, an inaccurate estimate will be got. A straightforward remedy is to delete the $\mathbf{u}_1$ whose corresponding ST window is not sparse enough, i.e., $1 - (\sigma_2/\sigma_1) > ebs$ where ebs denotes a real number close to 1. In this way, we only use the reliable cluster samples and we can get a better estimate reasonably. It should be stated that the scaling and permutation uncertainty exists in the estimated mixing matrix. The algorithm can be summarized in Table I.

B. Two-Layer Sparsity Model

Most of the energy of speech signals simultaneously focuses on the low frequency band, which will greatly reduce the sparsity and result in a degraded separation performance. From

TABLE I
ALGORITHM 1: ESTIMATE THE MIXING MATRIX

1. Compute the sparse coefficients of the two mixture vectors by STFT, and obtain $\tilde{\mathbf{X}}_i,\ i=1,2$.

2. Set $\mathbf{v}$ to null set, and assign ebs a number close to 1. For every ST window, remove the mean of $\tilde{\mathbf{X}}_i, i=1,2$ so that they have zero mean. We exploit SVD to $\mathbf{R}_{\tilde{\mathbf{X}}}$. If $1-\sigma_2/\sigma_1 > ebs$, then add $\mathbf{u}_1$ to $\mathbf{v}=[\mathbf{v},\mathbf{u}_1]$.

3. Run the K-means clustering algorithm to all the columns of $\mathbf{v}$ until convergence, and compute the column vectors of the estimated mixing matrix as the final cluster centers.

this fact, DCT dictionary, i.e., the inverse of DCT matrix, whose atom represents a digital frequency, is not a good choice to conduct separation. Inspired by the analysis about sparsity above, we desire to design a dictionary on which the sources are sparser, namely, different sources occupy as different dictionary bands as possible. If we can achieve this goal, we believe the separation result is preferable.

Trained dictionary from dictionary learning method is utilized here because of its full adaptability. Unlike the fixed dictionary, the trained dictionary is flexible and if we use specific speakers' speech for training, the specific speakers' features can be captured and reflected in the atoms of trained dictionary. K-SVD ([15]–[17]) is a good choice because it's highly structured. The K-SVD algorithm accepts an initial overcomplete dictionary, a number of iterations, and a set of training signals arranged as the columns of the matrix $\mathbf{T}$. The algorithm aims to iteratively improve the dictionary $\mathbf{D}$ to achieve sparser representations of the signals in $\mathbf{T}$, by solving the optimization problem:

$$\min_{\mathbf{D},\mathbf{\Gamma}} \|\mathbf{T}-\mathbf{D}\mathbf{\Gamma}\|_F^2 \quad s.t. \quad \|\boldsymbol{\gamma}_k\|_0 \le q,\ \forall k, \tag{17}$$

where $\boldsymbol{\gamma}_k$ is the kth column of $\mathbf{\Gamma}$. The K-SVD algorithm involves two basic steps, which together constitute the algorithm iteration: 1) the signals in $\mathbf{T}$ are sparse-coded given the current dictionary estimate, producing the sparse representations matrix $\mathbf{\Gamma}$; and 2) the dictionary atoms are updated given the current sparse representations. The sparse-coding part is commonly implemented using OMP. The dictionary update is performed one atom at a time. Every time the algorithm optimizes the target function for each atom individually while keeping the rest fixed. The efficient version of K-SVD [16] is exploited in this work.

We assume that before separating sources we have some speech samples as training data. From this sense, we would rather call the procedure semi-blind source separation. Instead of directly training K-SVD dictionary on the samples, we train dictionary on the low frequency of the data, i.e., we use a low-pass filter to obtain the low frequency components, and then use them as training samples to train dictionary. This process is reasonable: 1) by training on the low frequency component, it is likely to focus attention on scattering the original intensive distribution of the low frequency band expansion coefficients, which is consistent with the second meaning of "sparsity"; 2) most of the energy focuses on the low frequency part, so the low frequency component part can approximate the sample to some extent.

The dictionary trained above does not provide a complete decompose of a speech because it can only approximate the low frequency part. DCT dictionary is utilized to fill the gap, which makes up the two-layer sparsity model. We will demonstrate that this model is superior to the directly trained K-SVD dictionary or DCT dictionary.

C. Dictionary Based on the Two-Layer Sparsity Model

We define some notations here: $\mathbf{B}_i$ is the overcomplete dictionary trained with K-SVD only using the low frequency part of training data, $\mathbf{\Phi}$ is DCT dictionary, $\mathbf{c}_i$ and $\boldsymbol{\alpha}_i$ are the sparse coefficientHs respectively,

$$\mathbf{D}_i = [\mathbf{B}_i|\mathbf{\Phi}], i=1,2,3 \tag{18}$$

concatenates two dictionaries (we call it two-layer dictionary in the remaining sections) and

$$\mathbf{g}_i = \begin{bmatrix} \mathbf{c}_i \\ \boldsymbol{\alpha}_i \end{bmatrix}, i=1,2,3 \tag{19}$$

concatenates two coefficients.

The two-layer sparsity model dictionary can be summarized as follows.

$$\mathbf{f}_i = \mathbf{B}_i\mathbf{c}_i + \mathbf{\Phi}\boldsymbol{\alpha}_i = \mathbf{D}_i\mathbf{g}_i, \quad i=1,2,3, \tag{20}$$

where $\mathbf{f}_i$ is the ith source signal vector defined in Section II-A.

If all the sources share one dictionary, we call it multichannel dictionary. Under this circumstance, all of the samples are used to train a single dictionary. So, we can express $\mathbf{D}$ in (6) as follows:

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{D}_0 \end{bmatrix}, \mathbf{D}_1=\mathbf{D}_2=\mathbf{D}_3=\mathbf{D}_0. \tag{21}$$

In fact, if we know three speakers' order in advance, we can also use the so-called monochannel dictionary [4], [30] training strategy whose sole difference is that it train every dictionary of every speaker respectively, namely, we train the ith speaker's dictionary only using the corresponding ith samples as training data and finally we get three different dictionaries corresponding to three different speakers, i.e.,

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{D}_3 \end{bmatrix}, \mathbf{D}_1 \ne \mathbf{D}_2 \ne \mathbf{D}_3. \tag{22}$$

In this formulation, we assume the order is in sequence, i.e., $\mathbf{D}_i$ is corresponding to the i th source. We summarize the dictionary training algorithm in Table II.

If the dictionary $\mathbf{D}_i$ has been trained, we can use them to design $\mathbf{D}$ according to (21) or (22).

D. Algorithm Summary

Now we can summarize the entire underdetermined BSS algorithm in Table III.

TABLE II
ALGORITHM 2: TRAINING TWO-LAYER DICTIONARY

1. Initialize the dictionary training parameters including: the number of training sentences, sparse coding target, dictionary size, number of iterations, initial dictionary. Obtain different training data according to different training strategies: monochannel or multichannel dictionary.

a). If monochannel two-layer dictionary is chosen, we use every speaker's samples as training data and train three different dictionaries.

b). If multichannel two-layer dictionary is chosen, we consider all speakers' samples as training data.

2. For all training samples (namely all columns of training data), extract the low frequency part and use them as the new training data.

3. Employ K-SVD to obtain the source-specific dictionary.

4. Concatenate the DCT dictionary and K-SVD dictionary to obtain $\mathbf{D}_i$.

TABLE III
ALGORITHM 3: BLIND SOURCE SEPARATING BASED ON CS FRAMEWORK

1. Estimate the mixing matrix $\mathbf{A}$ using algorithm 1 and then obtain $\mathbf{M}$ using (3).

2. Use Algorithm 2 to train two-layer dictionary $\mathbf{D}$.

3. Obtain the product of $\mathbf{M}$ and $\mathbf{D}$ marked as $\tilde{\mathbf{M}}$ using (7).

4. Use the BP algorithm to find the sparsest coefficients $\mathbf{g}$ by (9).

5. Compute the signal in original (time) domain using (6).

6. Split the resultant single vector into multiple source vectors using $\mathbf{f} = [\mathbf{f}_1^T, \mathbf{f}_2^T, \mathbf{f}_3^T]^T$.

IV. EXPERIMENTAL RESULT

In this section we will firstly give a simulation to demonstrate the advantage of the proposed estimating mixing matrix algorithm. Afterwards, we will compare the proposed method with some recent methods. At last, we will demonstrate the superiority of two-layer dictionary compared to the DCT dictionary and some other recently proposed trained dictionaries.

The three speakers' speech database is downloaded from the website http://www.speech.cs.cmu.edu/cmu_arctic/. For every speaker, we have 10 sentences for testing and 20 sentences for training. Every sentence has a length of 3–5 seconds. The training sentences and testing sentences are absolutely different, i.e., if a sentence is chosen to be training sentence, it cannot be used for test. For every experiment, which testing sentences are chosen is based on the detail requirements.

A. Estimating the Mixing Matrix

1) The Performance for Different Mixing Matrices: We compare our proposed algorithm 1 with the method in [12] (is essentially DUET, marked as "duet") and [6] (marked as "duetiter"). In this experiment, we perform the above algorithms in the following experimental conditions: we have 3 sentences and every time we mix them using the following mixing matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ \tan(45^\circ - i^\circ) & \tan 45^\circ & \tan(45^\circ + i^\circ) \end{bmatrix}, \quad i = 10, 11, \ldots, 44.$$

As i increases from 10° to 44°, the similarity between columns of the mixing matrix becomes smaller. We estimate the mixing matrix and calculate the performance, which is measured by the computational time (the total time spent in 35 experiments) and the mixing error ratios (MER) criterion [30] from SiSEC 2010 [31].

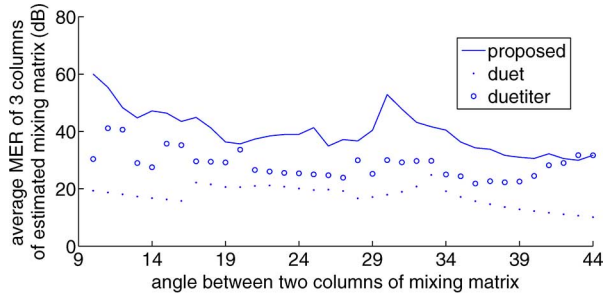


Fig. 2. The MER (measured by dB) of some methods: proposed algorithm 1, "duet", and "duetiter".

TABLE IV
COMPARISON OF TIME COST OF PROPOSED ALGORITHM 1 AND ALGORITHM IN [6], [12]. IN THE COMPUTATION, MATLAB IS USED WITH A PC, AN INTEL CORE2 QUAD CPU (2.80 GHZ) AND A WINDOWS 7 OS

proposed	algorithm in [12]	algorithm in [6]
33s	1174s	10740s

The results are shown in Fig. 2 and Table IV. From the result, one can see that the proposed method is always more accurate than the method in [12] and takes less time than the method in [12] and [6]. The method in [12] is only appropriate for the sparse source case. Otherwise the method will degrade seriously. The method in [6] can improve the accuracy but increases the computational cost. Our proposed estimating method allows for deleting the insufficiently sparse points before clustering (see Fig. 1). It is evidently more accurate than the method in [12]. Moreover, it is computationally less expensive than the method in [6].

2) The Performance for Different Sources: In order to show that our method works well for mixtures of different sentences, we perform one experiment in the following conditions: we once arbitrarily choose three different sentences from the testing sentences and mix them using the mixing matrix

$$\mathbf{A} = \begin{bmatrix} 0.6118 & 0.9648 & 0.2360 \\ 0.7910 & 0.2629 & 0.9718 \end{bmatrix}.$$

For every mix we use proposed algorithm and algorithm proposed in [12] and [6] to estimate the mixing matrix. We repeat this experiment 5 times and calculate the average MER of every column. The results are shown in Table V, demonstrating that the proposed algorithm is better than the other studied methods for different sources.

TABLE V
COMPARISON OF MER OF PROPOSED ALGORITHM 1 AND ALGORITHM IN [6], [12]

Average MER of 5 experiments for every column			
	proposed	algorithm in [12]	algorithm in [6]
Col1	32.5606	16.4523	16.1662
Col2	39.9756	6.5644	12.8036
Col3	34.3843	31.4816	26.6869

TABLE VI
THE PARAMETERS FOR K-SVD

number of training data	20 speeches for every speaker
window length	1024
low pass filter	1/8 (i.e., [ones(128,1),zeros(896,1))])
Tdata (sparsity coding target)	100
dictionary size	1024*3072
iteration number	10
window overlap	50% (i.e. 512)
sampling rate	16000

B. Comparison of Separation Performance Between the Proposed Algorithm 3 and Some Recent Algorithms

1) The Performance for Different Mixing Matrices: In this experiment, we compare our proposed algorithm 3 with some recent algorithms such as algorithm proposed in [1] (marked as "duet"), [6] (marked as "duetiter"), and [7] (marked as "ovlpduet"). The ideal DUET (marked as "duetideal") uses the true mixing matrix to conduct separation. We also use some other dictionaries such as DCT (marked as "dct") and K-SVD (marked as "ksvdonly") dictionary to replace our proposed multichannel two-layer dictionary. In this experiment, 3 testing sentences are arbitrarily chosen every time and we mix them using the same mixing matrix as in Section IV-A-1 except that the step size of angle is 2°. We repeat it 5 times and calculate the average performance of 5 times for different angles. The multichannel dictionary is used here and the parameters for K-SVD dictionary are shown in Table VI. For our proposed algorithm, the mixing matrix is not known a priori and we use the proposed algorithm 1 to estimate it. Then we use the estimated matrix for separation. We measure the performance using signal to distortion ratio (SDR), SIR, and signal to artifacts ratio (SAR) from SiSEC 2010 [31], [32]. The result is shown in Fig. 3.

From Fig. 3, one can draw a conclusion that when the angle between columns increases, i.e., the similarity of columns decreases, our proposed method performs better. When the angle increases to 12°, the SDR of our proposed method exceeds the ideal DUET, which can be regarded as the upper bound of DUET [1] and even its improved versions [6], [7]. This is because the weak sparsity of speech signals, which makes the binary masking method not so effective. The SIR of proposed method becomes better when the angle increases, and our method is more stable. Nevertheless the SAR of the proposed method is always the best among the studied algorithms when the angle increases from 10° to 40°. From the figure, one can also confirm that our proposed method is always better than DCT and K-SVD dictionary. Furthermore, it is less affected by the estimating mixing matrix stage.

2) The Performance for Different Sources: To further demonstrate the superiority of the proposed algorithm, we compare it with other studied algorithms in the following conditions: we once arbitrarily choose three different sentences from testing sentences and mix them with the same mixing matrix as in Section IV-A-2. Again, we repeat for 5 times. The result is shown in Table VII. From Table VII, one can see that our proposed method is always better for different sources than the other studied methods.

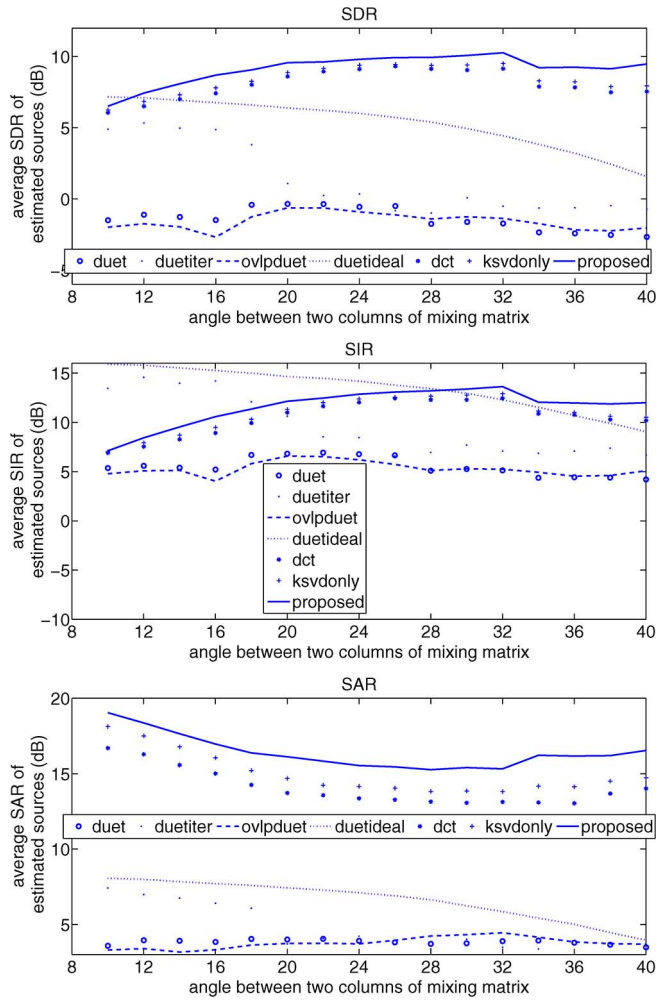


Fig. 3. The SDR, SIR, SAR (measured by dB) of 7 methods: multichannel dictionary based on proposed model, "dct", "ksvdonly", "duet", "duetideal", "duetiter" and "ovlpduet".

C. Comparison Between Dictionary Based on the Proposed Model and DCT Dictionary, K-SVD Dictionary, PCA Dictionary, GAD

In this experiment, we study the performance of using different types of dictionary $\mathbf{D}_i$: monochannel two-layer dictionary, multichannel two-layer dictionary, DCT dictionary, PCA dictionary, K-SVD dictionary and GAD. In order to distinguish the K-SVD dictionary from the proposed dictionary, we call this K-SVD dictionary "K-SVD only". The parameters of "K-SVD only" are the same as K-SVD in the proposed algorithm except that a low-pass filtering is not used for the training data. To be fair, we assume the mixing matrix is the same as the Section IV-A-2 and is known prior. The result is shown in Fig. 4. In order to save simulation time, we set the window length to 400, and we have 20 training sentences and 10 testing sentences for every speaker as well. We arbitrary choose 3 different testing

TABLE VII
COMPARISON OF SDR, SIR AND SAR FOR DIFFERENT SOURCES

Average SDR(dB) of 5 experiments			
method	Source1	Source2	Source3
proposed	8.0868	17.7373	12.4996
dct	3.1930	14.3660	10.1542
ksvdonly	4.9738	15.8018	11.3307
duet	-12.6961	8.1667	-0.3085
duetiter	-8.0876	8.8332	2.6568
ovlpduet	-12.7126	7.4321	-2.3432
duetideal	2.6693	10.9246	0.1992
Average SIR(dB) of 5 experiments			
method	Source1	Source2	Source3
proposed	13.2140	19.5834	13.7599
dct	5.7843	16.9233	12.7526
ksvdonly	8.6525	18.2549	13.5059
duet	-7.8259	9.9026	15.9824
duetiter	-2.0453	15.2677	16.2811
ovlpduet	-9.8183	10.7482	14.1268
duetideal	8.2099	21.9263	10.4571
Average SAR(dB) of 5 experiments			
method	Source1	Source2	Source3
proposed	10.0665	22.5552	18.7037
dct	7.9544	17.9935	13.8596
ksvdonly	8.2487	19.6394	15.5719
duet	-0.2060	17.2703	-0.0571
duetiter	-0.3478	13.9087	2.9653
ovlpduet	4.0599	13.1783	-2.0707
duetideal	4.8734	11.3222	1.0385

sentences of the three speakers every time and measure the performance using the average SDR, SIR and SAR of 5 times.

The DCT dictionary performs well if the sources are sparse. But if the sources are not sparse enough, the performance tends to deteriorate. In comparison, the dictionary based on the proposed model, merging both advantages of DCT dictionary and K-SVD dictionary, is efficient and adaptive. From this result one can see that the two-layer sparsity model dictionary is consistently better as compared to DCT dictionary.

From Fig. 4, "K-SVD only" dictionary performs better than the PCA and GAD dictionaries. This is due to the fact that the trained K-SVD dictionary is highly structured [17]. Our experiment provided evidence that the dictionary designed based on the two-layer sparsity model, benefiting from the high adaptability of K-SVD and the appropriate two-layer model assumption, exceeds others.

The monochannel two-layer dictionary gets better result than multichannel two-layer dictionary. It's reasonable because every speaker's dictionary is trained using specific source samples. Obviously the monochannel dictionary, which is source-dependent, can capture sources' features more easily and accurately. Note that the sources' sequence is usually easy to get, so in most cases, the monochannel dictionary is available.

D. Discussion on the Choice of Some Parameters

There are a number of parameters to choose in the separation using algorithm 2, including the parameters in K-SVD, training strategy (monochannel or multichannel dictionary), window length l and overlap. We list some primary ones here.

Generally speaking, more samples used for training can obtain a more valid dictionary. But when the number of samples reaches a threshold, it would lead to little enhancement but huge time cost for training. The choice of window length depends on the

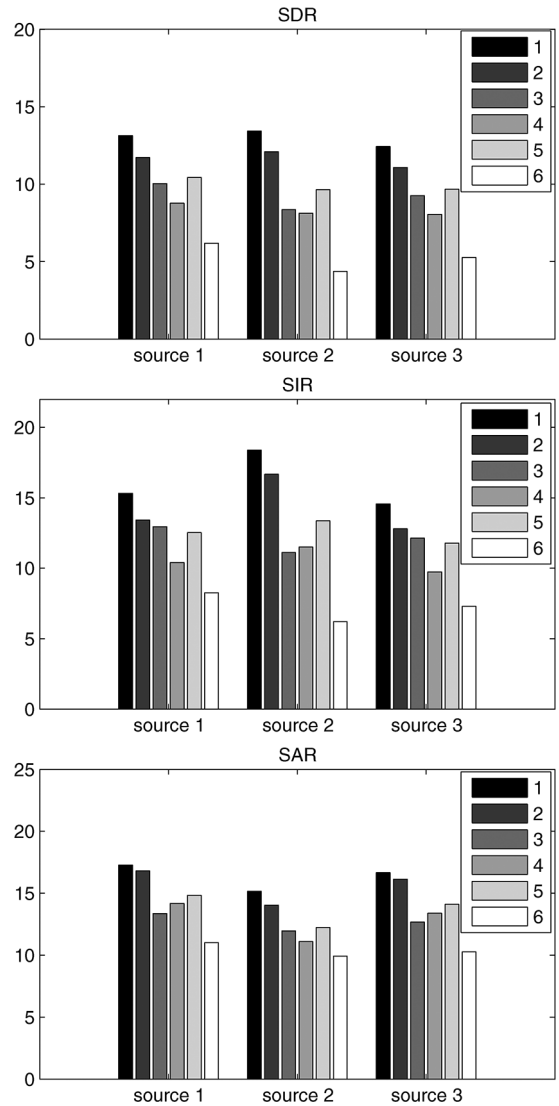


Fig. 4. The average SDR, SIR, SAR (measured by dB) of six methods: monochannel dictionary based on proposed model (mark as "1"), multichannel dictionary based on proposed model (mark as "2"), DCT dictionary (mark as "3"), PCA dictionary (mark as "4"), "K-SVD only" dictionary (mark as "5") and GAD (mark as "6").

sampling rate of speech signal, empirically 1024. The low-pass filter is chosen according to the source that the low frequency part can contain about 60%–80% energy of a speech signal. We usually choose overcomplete DCT dictionary as the initial dictionary to reduce the number of iterations. Taking training time into account, we usually use the fastest implementation of OMP [16]. It is clear that a larger overlap can give a better performance, but it would be computationally more expensive to process every speech signal. We usually exploit 50% overlap. If the sequence of sources can be determined before separation by any means, the monochannel dictionary is expected.

V. CONCLUSION

In this work we proposed a CS approach to underdetermined BSS which contains two stages. Two dictionaries based on a two-layer sparsity model are designed in the second stage. Experiments showed that our approach can estimate the mixing matrix more precisely in the first stage and get a better separation

performance than DUET and even some of its recent improved versions in the second stage. Furthermore, it is obvious that the proposed dictionary performs consistently better than DCT dictionary and some recently presented dictionaries that are proved to be effective in many other applications. Overall, the proposed model improves blind source separation performance visibly. Note that the proposed method and other methods based on dictionary learning need training while DUET and its improvements do not.

REFERENCES

[1] A. Jourjine, S. Rickard, and Ö. Yilmaz, "Blind separation of disjoint orthogonal signals: Demixing N sources from 2 mixtures," in *Proc. IEEE Int. Conf. Acoustic, Speech, Signal Process. (ICASSP)*, 2000, pp. 2985–2988.

[2] Y. Li, S. Amari, A. Cichocki, W. C. Ho. Daniel, and S. Xie, "Underdetermined blind source separation based on sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 423–436, Feb. 2006.

[3] B. V. Gowreesunker and A. H. Tewfik, "Two improved sparse decomposition method for blind source separation," in *Proc. Ind. Compon. Anal. Signal Separat. (ICA) LNCS*, 2007, vol. 4666, pp. 365–372.

[4] B. V. Gowreesunker and A. H. Tewfik, "Blind source separation using monochannel overcomplete dictionaries," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2008, pp. 33–36.

[5] Ö. Yılmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 1830–1847, Jul. 2004.

[6] J. Han and B. Pardo, "Improving separation of harmonic sources with iterative estimation of spatial cues," in *Proc. IEEE Workshop Applicat. Signal Process. to Audio and Acoust. (WASPAA)*, 2009, pp. 77–80.

[7] Y. Lv and S. Li, "Underdetermined blind source separation of anechoic speech mixtures in the time-frequency masking," in *Proc. Int. Conf. Signal Process. (ICSP)*, 2008, pp. 22–25.

[8] E. J. Candès, "Compressive sampling," in *Proc. Int. Congr. Mathematicians (ICM)*, 2006.

[9] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[10] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 14–20, Mar. 2008.

[11] M. Fornasier and H. Rauhut, "Compressive Sensing," in *Handbook of Mathematical Methods in Imaging*, O. Scherzer, Ed. New York: Springer, 2011.

[12] T. Xu and W. Wang, "A compressed sensing approach for underdetermined blind audio source separation with sparse representation," in *Proc. IEEE Int. Workshop Statist. Signal Process.*, 2009, pp. 493–496.

[13] T. Xu and W. Wang, "Methods for learning adaptive dictionary in underdetermined speech separation," in *Proc. IEEE Int. Workshop Mach. Learn. for Signal Process.*, 2011, pp. 1–6.

[14] T. Xu and W. Wang, "A block-based compressed sensing method for underdetermined blind speech separation incorporating binary mask," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2010, pp. 2022–2025.

[15] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representations," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[16] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," Technion-Israel Insti. of Technol., 2008, Tech. Rep..

[17] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning sparse dictionaries for sparse signal approximation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1553–1564, Mar. 2010.

[18] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1945–1959, Dec. 2005.

[19] N. Katsumata and Y. Matsuyama, "Similar-image retrieval systems using ICA and PCA bases," in *Proc. Int. Joint Conf. Neural Netw.*, 2005, pp. 1229–1234.

[20] M. G. Jafari and M. D. Plumbley, "Fast dictionary learning for sparse representations of speech signals," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 5, pp. 1025–1031, Sep. 2011.

[21] T. Blumensath and M. E. Davies, "Compressed sensing and source separation," in *Proc. Int. Conf. Ind. Compon. Anal. Blind Source Separat. (ICA)*, 2007, pp. 341–348.

[22] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1999.

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

[24] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constructive Approx.*, vol. 13, no. 1, pp. 57–98, 1997.

[25] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching Pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Conf. Rec. 27th Asilomar Conf. Signals, Syst., Comput. (ACSSC)*, 1993, pp. 40–44.

[26] S. Mallat and Z. Zhang, "Matching pursuits with TF dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[27] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600–616, Mar. 1997.

[28] D. L. Donoho, Y. Tsaig, I. Drori, and J. L. Starck, Department of Statistics, Stanford University, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," Tech. Rep. 2006-2, 2006.

[29] N. Hurley and S. Rickard, "Comparing measures of sparsity," *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4723–4741, Dec. 2009.

[30] E. Vincent, S. Araki, and P. Bofill, "The 2008 signal separation evaluation campaign: A community-based approach to large-scale evaluation," in *Proc. Int. Conf. Ind. Compon. Anal. Signal Separat. (ICA)*, 2009, pp. 734–741.

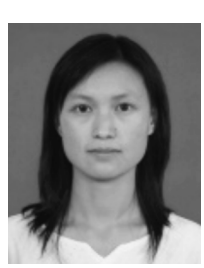
[31] [Online]. Available: http://sisec.wiki.irisa.fr/tiki-index.php

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

Guangzhao Bao (S'12) received the B.E. degree in electronics and information engineering from the University of Science and Technology of China, Hefei, China, in 2010. Currently, he is working towards the Ph.D. degree in the University of Science and Technology of China, Hefei, China. His research interests are in array signal processing, especially the blind source separation and sparse representation.

Zhongfu Ye received the B.E. and M.S. degrees in electronics and information engineering from the Hefei University of Technology, Hefei, China, in 1982 and 1986, respectively, and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 1995. He is currently a Professor of the University of Science and Technology of China. His current research interests are in statistical and array signal processing and image processing.

Xu Xu received the B.E. degree in electronic engineering from the Hefei University of Technology, Hefei, China, in 1997 and the M.S. degree in communication and information system from the University of Science and Technology of China, Hefei, China, in 2000. Since 2000, she has worked as a teacher at the Department of Electronic Engineering and Information Science of the University of Science and Technology of China. Her current research interests are in array signal processing for wireless communication.



Yingyue Zhou received the B.E. degree in biomedical engineering from the Southwest University of Science and Technology, Mianyang, China, in 2005 and the M.S. degree in biomedical engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2008. She is now a Ph.D. candidate of the Department of Electronic Engineering and Information Science of USTC. Her research interest covers image processing and analysis, especially the image recovery and the applications of sparse and redundant representation.