Blind Sources Separation Based on Short-Time Discrete Cosine Transforms

Zeng Yanni[1,2], Zhang Yujie[1*], Qi Rui[1,3]

[1]*School of Mathematics and Physics, China University of Geosciences, Wuhan, China*

2 Faculty of Statistics and Applied Mathematics, Hubei University of Economics, Wuhan, China

3School of Science, Naval University of Engineering, Wuhan , PR China

Received 1 November 2014, www.cmnt.lv

Abstract

This paper presents a sparse blind source separation method which uses short-time discrete cosine transform (STDCT) to obtain the transformed domain information from a set of linear instantaneous mixtures of these sources. Unlike short-time Fourier transform (STFT) determining the single source point or area by using the ratio of the real and imaginary parts, we remove these points which are away from the mean direction of the cluster using the orthogonal distance between the point and the line. For the clustering part, we use, in this paper, an algorithm inspired from K-means. The final algorithm is easy to extend for any number of sources. Because the STDCT is a Fourier-related transform similar to the STFT, which using only real numbers, so it reduces the computer cost on clustering and improves the algorithm accuracy. Experimental results are provided to evaluate the performance of the proposed algorithm through comparing with STFT from the normalized mean-square error (NMSE) and signal-to-noise ratio (SNR).

Keywords: Blind source separation, Short-time discrete cosine transforms, K-means cluster, Orthogonal distance, Sparsity

1 Introduction

Blind source separation (BSS) consists in retrieving unknown source signals from a set of their mixtures without any knowledge about the mixing system or the signals [1-2]. The instantaneous linear mixture blind source separation (BSS) problem can be described as [1-2]:

$$\mathbf{y}(t) = \mathbf{A}\mathbf{s}(t), \quad t = 1, 2, \cdots, n \quad (1)$$

where $\mathbf{y}(t) = [y_1(t), y_2(t), \cdots, y_m(t)]^T$ are the observed signals and $\mathbf{s}(t) = [s_1(t), s_2(t), \cdots, s_n(t)]^T$ are the unknown original source signals. t is the discrete time sequence and T is the transpose operator. $\mathbf{A} = (a_{ij})_{m\times n}$ is an unknown full row rank mixture matrix. The main objective of BSS is to estimate the mixture matrix $\mathbf{A}$ and the sources $\mathbf{s}$.

Many algorithms have been developed for the linear instantaneous mixtures and are based on independent component analysis (ICA). They assume the sources to be random stationary statistically independent signals and the number of mixtures is more than or equal to the number of sources which called overdetermined and determined BSS. However, in practical situations, the number of mixed signals may be less than the number of sources, and cases like this are called underdetermined BSS (UBSS).

The UBSS problem is generally more difficult than the determined and overdetermined BSS problem. Most conventional underdetermined BSS algorithms were developed based on the sparsity assumption of the sources [3-10], which called sparse component analysis (SCA). For sparse source signals, the mixture matrix can be obtained by a clustering of observed samples, and then the direction of the modulus of the observed signals will be the same as those of the column vectors of the mixing matrix. A signal is said to be sparse in the temporal domain when most of the signal amplitude are zero during the time period, and only a few number have significant values (active). If only one signal is active at a time, we call the sources are sufficient sparse. However, in practice, the natural signals are not very sparse in the time domain like speech signals which are sparser in the frequency domain than in the time domain [4]. Hence if we transform the time domain signal into the transformed domain using a linear transform, the sparsity can be utilized to obtain the sources estimation from their mixtures. Recently, some algorithms have been proposed to achieve the sparsity in transformed domain, such as wavelet packet transform (WPT) [5-6] or short-time Fourier transform (STFT) [7-9]. Although STFT does not introduce interference terms, it includes real and imaginary parts which increase the data, and WPT involves the choice of the decomposition level, they result in high computational cost. In contrast, DCT is a Fourier-related transform similar to discrete Fourier transform (DFT), but using only real numbers which reduces the computational efficiency [10]. As same as DFT, for obtaining the local time frequency information, we use short-time DCT (STDCT) for BSS to instead of STFT and WPT in this paper.

* ***Corresponding author's*** e-mail: rubycock@163.com

A UBSS is often solved by two-stage approach. Firstly, a mixing matrix is estimated given only the observed signals. In this paper, STDCT is used to obtain the sparse signals and then estimated the mixing matrix using an improved K-means cluster method. Secondly, the underlying sources are retrieved given the observed signals, the estimated mixing matrix. In the first stage, among the trends which emerge from these papers, the following things should especially be mentioned. A first set of methods requires that the sources be very sparse in the time domain or the transformed domain [3, 5]. The second set of methods relaxes the requirement and requires that there exist time-frequency regions where only a single source is active for each source [9-11]. The main objective in all these algorithms is the detection of these domains or regions. Then conventional algorithms are used to estimate mixing matrix based on clustering algorithms such as the K-means algorithm [11] or hierarchical clustering algorithm [3], or time frequency ratio of mixtures (TIFROM) method [9-10]. In this paper we address the problem of estimation of mixing matrix from their instantaneous underdetermined mixtures only. Unlike STFT determining the single source point or area by using the ratio of the real and imaginary parts, we use the orthogonal distance between the point and the line to remove the points which are away from the mean direction of the cluster.

In this paper, we consider SCA problems in the underdetermined case, where the additional information compensating the limited number of sensors is the sparseness of the sources. It should be noted that this problem is quite general, since the sources could be not necessarily sparse in time domain. It would be sufficient to find a linear transformation (e.g. STDCT), in which the sources are sufficiently sparse. So we begin with two assumptions:

Assumption 1: There exist time domains (regions) or transformed domains (regions) where only one source occurs. That means the sources are not sufficient sparse, but exist sparse area.

Assumption 2: The columns of mixture matrix $\mathbf{A}$ are normalized to have unit l_2-norm, respectively. i.e.

$$\sum_{i=1}^{m} a_{ij}^2 = 1,\ j = 1, \cdots, n \ [13].$$

It is well known that knowing the mixing matrix does not directly result in the recovery of the sources [11]. In this paper, we only consider the problem of the estimation of the mixing matrix. Firstly, we use STDCT to get the sparse signals, and then estimate the mixture matrix by cluster analysis. At the same time, we proposed two steps to remove those points which away from the cluster center and/or have very small values. Finally, we give three experiments to research the performances of our algorithm comparing with STFT.

This paper is organized as follows. Section 2 formulates the problem that we are addressing and introduces STDCT. Section 3 presents an improved K-means clustering algorithm for estimating the mixing matrix. Section 4 gives some simulation results and finally conclusions are drawn in Section 5.

2 Problem formulation and STDCT

2.1. SCA MODEL

For SCA, model (1) can be written as:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) = \begin{pmatrix} a_{11} \\ \vdots \\ a_{m1} \end{pmatrix} s_1(t) + \cdots + \begin{pmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{pmatrix} s_n(t), \quad t = 1, 2, \cdots, N. \quad (2)$$

For a fixed time t, if only one source is active, without loss of generality, suppose that $s_i(t)$ has significant value, i.e., $s_i(t) \neq 0$ and $s_j(t) = 0,\ j \neq i, \quad j = 1, 2, \cdots, n$. Eq. (2) then becomes

$$\begin{pmatrix} x_1(t) \\ \vdots \\ x_m(t) \end{pmatrix} = \begin{pmatrix} a_{1i} \\ \vdots \\ a_{mi} \end{pmatrix} s_i(t) \ . \quad (3)$$

Obviously, formula (3) means that all columns of $\mathbf{A}$ are the hyperline directions of the modulus of the observation vectors. In other words, to estimate $\mathbf{A}$, we only need to find these directions by solving adaptive clustering problem.

The essence of the sparse approach is the identification of directions from the observation signals. And the higher sparsity is a requirement for good estimation of mixing matrix. In this case a possible approach is to look for a linear transform T such that the new representation of the data is sparser [4]. Then Eq. (2) can be rewritten as

$$\mathbf{X} = T(\mathbf{y}) = \mathbf{A}T(\mathbf{s}) = \mathbf{AS} \quad (4)$$

where $\mathbf{X}$ and $\mathbf{S}$ are, respectively, the transformed coefficients of the mixtures and sources. For decreasing the computation cost, STDCT has been used in here. Because STDCT is a Fourier-related transform similar to the discrete Fourier transform, but using only real numbers.

2.2. SHORT TIME DISCRETE COSINE TRANSFORM

STDCT expresses a function or a signal in terms of a sum of sinusoids with different frequencies and amplitudes. It is important to a large number of applications in science and engineering, such as compression of audio and images, to spectral methods for numerical solution of partial differential equations. We express the unitary STDCT of each observed signal [14]:

$$d_i^j(k) = w(k) \sum_{t=1}^{l} h(t) y_i[t + (j-1) \times l] \cos\left(\frac{\pi(2t-1)(k-1)}{2l} \right),$$
$$k = 1, 2, \cdots, l, \quad i = 1, 2, \cdots, m,\ j = 1, 2, \cdots, M \ , \quad (5)$$

where

$$w(k) = \begin{cases} \dfrac{1}{\sqrt{l}}, & k = 1 \\ \sqrt{\dfrac{2}{l}}, & 2 \le k \le l \end{cases} , \quad (6)$$

$h(t)$ is a shifted windowing function, in this paper, we use Hanning window. l is the window length. $M = \lfloor N/l \rfloor$ is the number of the window, where $\lfloor \bullet \rfloor$ stands for rounding down to the nearest whole number. Let $x_i = [d_i^1, d_i^2, \cdots, d_i^M]$ and $\mathbf{X} = [x_1, x_2, \cdots, x_m]^T$.

The STDCT steps are described follow:

Step 1: The mixtures were processed in frames of length l samples and they were multiplied by a Hanning window. Consecutive frames are with an overlap of $l-d$.

Step 2: Each frame was transformed with a DCT of length l, and then form the new signals $\mathbf{X}$.

Note that the STDCT is a linear transform, so when we use STDCT to the observed signals, the source signals have also been transformed to the transformed domain. The estimation of the mixing matrix is performed using the transformed coefficients $\mathbf{X}$ of the observed signals instead of the observed signals $\mathbf{y}$.

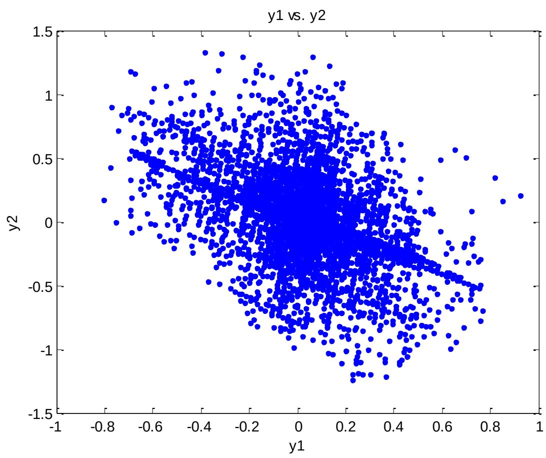
2.3. COMPARISON WITH DIFFERENT LINEAR TRANSFORMATIONS

We begin with an example of two speech signals in different linear transformations. The considered two sources are from the experiment "FourVoices_src" in [15], each source has 10000 samples. The mixing matrix is followed

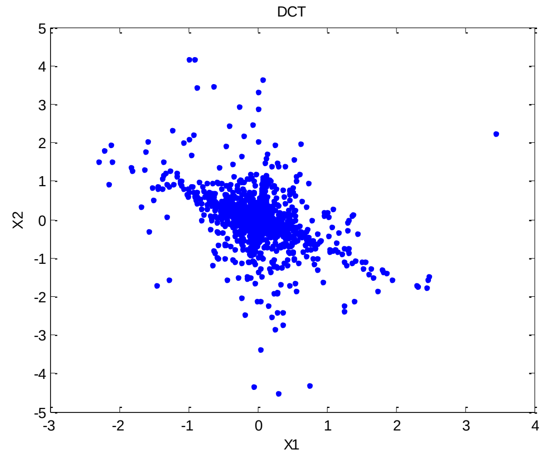
$$\mathbf{A} = \begin{bmatrix} -0.1079 & -0.8069 \\ -0.9942 & 0.5907 \end{bmatrix}.$$

Two observed signals are obtained by $\mathbf{y} = \mathbf{As}$. Fig. 1 gives the scatter plot $\mathbf{y}_1$ against $\mathbf{y}_2$ with no transform, DCT and STDCT. Fig. 1(a) gives the scatter plot $\mathbf{y}_1$ against $\mathbf{y}_2$ without transform, which presents a scatter plot of the observed signals showing a single big cloud. As can be seen, the different sources are indistinguishable. Then each mixture is DCT and the scatter plot of the transformed domain data is shown in Fig. 1(b), from which we cannot see noticeable line directions. So STDCT with Hanning window is used.

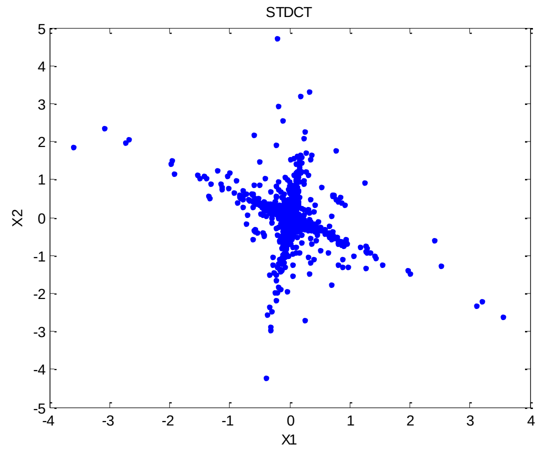
The scatter plot of $\mathbf{X}$ with $l = 2048$, $d = 614$ is showed in Fig. 1(c). In contrast, STDCT is obviously better than the scatter plot in time domain and DCT in this example. Almost all significant data are clustered along the two directions of the vectors.



(a) The observed signals $\mathbf{y}_1$ and $\mathbf{y}_2$



(b) The transformed domains of DCT



(c) The transformed domains of STDCT

FIGURE 1 Scatter plot $\mathbf{y}_1$ against $\mathbf{y}_2$ which mixed by two speech signals with different transforms.

Note that the length of the transformed domains with DCT is same as the observed signals, that are 10000 samples, but the length of the transformed domains by STDCT with $l = 2048$, $d = 614$ is 26624 samples. And if we use STFT with the same window, the length of the transformed domains (real and imaginary part) is 53248 samples, which is twice than the STDCT. In order to increase the estimation accuracy and improve the computation cost, we choose STDCT in this paper.

3 Proposed approach

As it is explained in the previous section, the main idea is to estimate the directions of the scatter plot of observations. If the sources are sufficient sparse in time domain or transformed domain, some clustering methods can be used to estimate the mixture matrix directly [4, 15]. Here, we research insufficient sparse case. Without lost of generality, we consider two-dimensional situation. And in the following sections, this idea can be easily generalized to more than two sources.

3.1. K-MEANS CLUSTER

Firstly, as same as [7,15], we standardized observed signals to upper unit semicircle, each line corresponds to one category on the unit semicircle. It contains two steps: mirror and normalization. Then use K-means cluster method to obtain the clustering centers which corresponding to the columns of the estimation mixture matrix. The estimation of the mixture matrix is

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0.2788 & -0.7005 \\ 0.8564 & 0.6346 \end{bmatrix}.$$

Using the data on section 2.2, we obtain the clustering centers and show them in Fig. 2. Red line represents the connection to the directions of clustering centers, and green line represents the directions corresponding to the column of A.

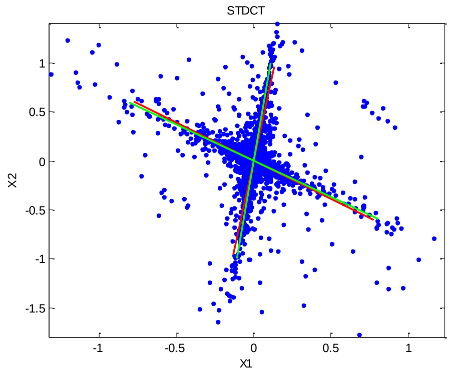


FIGURE 2 The scatter plot of the observed signals (blue points), the clustering centers (red lines) and the columns of A (green lines).

We use the normalized mean-square error (NMSE) (in dB) to evaluate the estimation quality. It is defined as:

$$\text{NMSE} = 10\log_{10}\left(\frac{\sum_{i,j}(a_{ij})^2}{\sum_{i,j}(\hat{a}_{ij} - a_{ij})^2}\right), \tag{7}$$

where $\hat{a}_{\alpha\beta}$ is (i, j) th element of the estimated matrix $\tilde{\mathbf{A}}$. If we cluster on the transformed domain directly, the NMSE is 15.1253dB.

From Fig. 2 and the NMSE, there are some distinction between red line and green line. For reducing the estimation error, an improved method is introduced in the following subsection.

For STFT, the absolute directions of real and imaginary parts of the observed signals are the same in the single source points. However, as shown previously, there are no real and imaginary parts in STDCT. We must remove the points which are away from the mean direction of the cluster and adjust the cluster center to the true directions of the mixture matrix only use the data of the STDCT.

Consider a data point (x_i, y_i), the line is $ax + by = 0$. Then the orthogonal distance between the point and the line is

$$d_i = \frac{|ax_i + by_i|}{\sqrt{a^2 + b^2}} \quad . \tag{8}$$

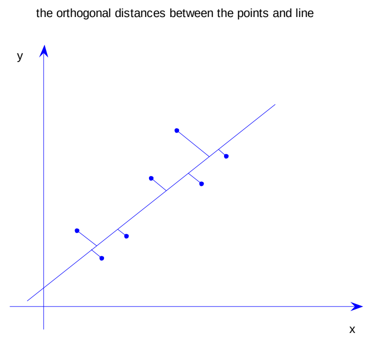


FIGURE 3 The orthogonal distances between the points and the line

Fig.3 gives the distances between the points and the line. From this figure, the farther the point is away from line, the greater the distance is. So we remove the points which are away from the mean direction of the cluster by comparing with a small positive constant ρ. In other words, ith sample is removed if $d_i > \rho$, where d_i is the orthogonal distance between ith sample point and the direction of ith sample in the cluster. This result of removing some points which are away from the cluster is described in Fig. 4.

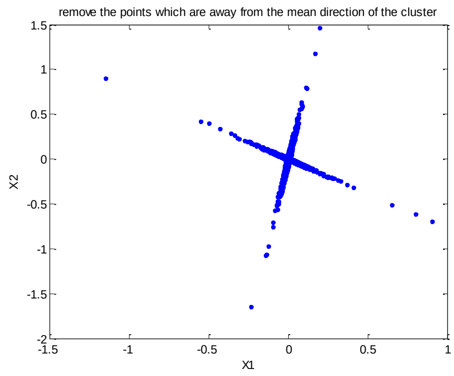


FIGURE 4 The scatter plot of removing some points which are away from the cluster

The mixing matrix estimation error can be further reduced by removing some samples which have very small values. These samples can be discarded after removing the points which away from cluster center. In other words, the samples satisfy

$\|\mathbf{X}(t)\| < c \max_t \left(\sum_{i=1}^{m} x_i(t)\right)$ are removed, where c is a small positive value.

It should be noted that the values of ρ and c need to be appropriately set. When ρ is too small, the samples are all too near to the cluster center, so it cannot adjust the cluster center to the true directions of the mixture matrix. And when ρ is too large, many unwanted points are reserved. When c is too small, the samples with small values cannot be removed clearly, When c is too large, the number of samples which are retained is less, which is bad for K-means cluster. In this paper, we study the dependence of the performance of STDCT to its parameters ρ and c.

3.2 N-DIMENSIONAL CASE

The approach stated above can be directly generalized to higher dimensions. Given a point $P=(p_1,p_2,\cdots,p_N)^T$ and a hyperline L in the N-dimensional space, of which the direction vector is $\mathbf{l}=(l_1,l_2,\cdots,l_N)^T\in\mathbb{R}^N$. The distance $d(P,\mathbf{l})$ from P to L is

$$d(P,\mathbf{l})=\langle P,P\rangle-\frac{\langle \mathbf{l},P\rangle^2}{\langle \mathbf{l},\mathbf{l}\rangle}\ . \tag{9}$$

We use Eq.(9) to compute the distance from point to line in higher dimensions. Consequently, when the distance from ith point to L larger than a constant ($d_i>\rho$), we remove ith point from the observed signals.

The procedure of the improved method is as follows:

Step 1. Perform STDCT of the observed signals $\mathbf{y}$ using Eq. (5), the transformed coefficients are $\mathbf{X}$.

Step 2. Standardize the signals $\mathbf{X}$ to $\tilde{\mathbf{X}}$ in the transformed domain.

Step 3. Split signals $\tilde{\mathbf{X}}$ to n clusters by K-means clustering, $\bar{\mathbf{x}}^i$ is the cluster center of ith class.

Step 4. Calculate d for each sample in ith class using Eq.(9), if $d>\rho$, remove this sample from ith cluster. After that, all the data which be leaved are $\hat{\mathbf{X}}$.

Step 5. Discard the sample point, if

$$\left\|\hat{\mathbf{X}}(t)\right\|<c\max_t\left(\sum_{i=1}^{m}\hat{x}_i(t)\right).$$

Step 6. Split the retained signals to n clusters by K-means clustering again.

Step 7. Repeat step 4-6 until convergence, i.e. the cluster centers are no longer changed.

The cluster centers are the estimation of the columns of mixture matrix. When we obtain the estimation $\tilde{\mathbf{A}}$ of the mixing matrix, if $m=n$ and $\mathbf{A}$ is invertible, the sources can be estimated by $\tilde{\mathbf{s}}=\tilde{\mathbf{A}}^{-1}\mathbf{y}$. Otherwise, the algorithms proposed in [10,12] be used for sources estimation in transformed domain, and then transform these estimation sources into time domain by inverse STDCT.

4 Experimental results

Simulations are performed on speech signals using the proposed algorithm. All of the experiments are repeated 50 times to obtain average performance. Each column of the mixing matrix can be randomly generated using the normal distribution and then is normalized to have unit l_2-norm. The other experimental conditions are: STDCT size 2048, Hanning window as the weighting function and overlap length is 614. Except for the section 4.1 for the parameters ρ and c research, $\rho=0.25$ and $c=0.1$.

The runtime is the CPU time. Our simulations are performed in MATLAB7.1 environment using an Intel Core2 Quad CPU 2.66GHz processor with 3.25G of memory, and under Microsoft Windows XP operating system.

To evaluate the estimation quality, signal-to-noise ratio (SNR) and NMSE are used. SNR is defined as:

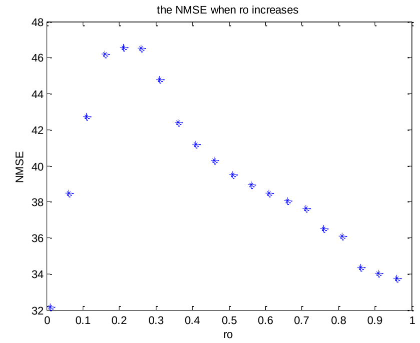
$$\text{SNR}=10\log_{10}\left(\frac{\|s\|}{\|s-\hat{s}\|}\right), \tag{10}$$

where s and $\hat{s}$ denote the actual sources and their estimation, respectively.

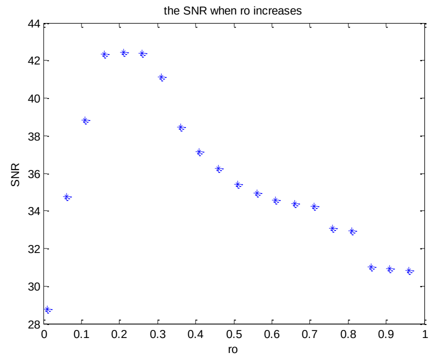
4.1 PARAMETER CHOOSING

In this simulation, $m=n=2$. The considered sources are from the experiment "FourVoices_src" in [15], each source has 10000 samples. We discuss the performance of the mixture matrix estimation and the sources retrieval fixing one of the parameters ρ and c. Fig. 5(a) and Fig. 5(b) illustrate the averaged NMSE and SNR when ρ increases with interval 0.05 from 0.01 to 1 and $c=0.1$, respectively. The numerical values on the x-axis in Fig. 5(a) and Fig. 5(b) denote the parameter ρ, while the numerical values on the y-axis represent the NMSE in Fig. 5(a) and the SNR Fig. 5(b), respectively.

As shown in Fig. 5(a) and Fig. 5(b), the numerical value of ρ cannot be too small or large. When ρ is too small, the samples are all too close to the cluster center, which cannot adjust the cluster center to the true directions of the mixture matrix. And when ρ is too large, many unwanted points are reserved which will influence the estimation accuracy. In this experiment, there is a best result when $\rho=0.21$. At the same time, the NMSE is consistent with the SNR when ρ changes. The larger the NMSE is, the better the matrix estimation is, and then the better the sources recovered is.



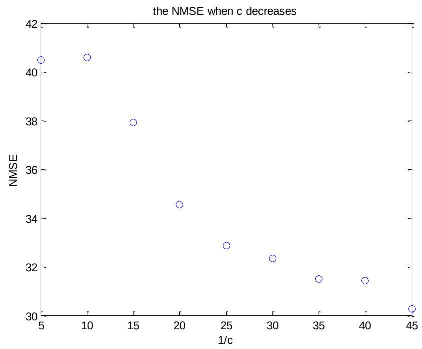
(a) The average NMSE



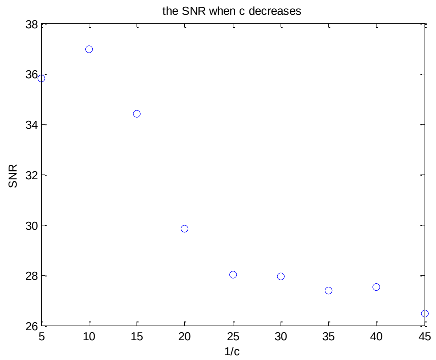
(b) The average SNR

FIGURE 5 The averaged NMSE and SNR when ρ increases with interval 0.05 from 0.01 to 0.1 and $c=0.1$.

Fig. 6(a) and Fig. 6(b) show the averaged NMSE and SNR when $1/c$ increases with interval 5 from 5 to 45 and $\rho=0.25$, respectively. The numerical values on the x-axis in Fig. 6(a) and Fig. 6(b) denote the parameter $1/c$, while the numerical values on the y-axis represent the NMSE in Fig. 6(a) and the SNR Fig. 6(b), respectively.



(a) The average NMSE



(b) The average SNR

FIGURE 6 The averaged NMSE and SNR when $1/c$ increases with interval 5 from 5 to 45 and $\rho = 0.25$.

As shown in Fig. 6(a) and Fig. 6(b), the numerical value of c cannot be too small or large. When c is too small, the samples with small values cannot be removed clearly, which will influence the estimation accuracy. When c is too large, the number of samples which are retained is less, which is bad for K-means cluster. In this experiment, there is a best result when $c = 0.1$. Meanwhile, the NMSE is consistent with the SNR when c changes. The larger the NMSE is, the better the matrix estimation is, and then the better the sources retrieved is.

4.2 THE COMPARATION WITH STFT

In this experiment, we use the STFT to obtain the sparse signals instead of STDCT, other conditions are same with 4.1. Because it has the real and imaginary part, we need joint the real and imaginary parts to cluster and obtain the mixing matrix estimation. This process increases the length of data, which increases the computation time and reduces the clustering accuracy in the cluster. The mixing matrix is followed

$$\mathbf{A} = \begin{bmatrix} -0.1079 & -0.8069 \\ -0.9942 & 0.5907 \end{bmatrix}.$$

With STDCT, the estimation of the mixing matrix $\mathbf{A}$ is obtained:

$$\hat{\mathbf{A}}_{STDCT} = \begin{bmatrix} 0.0970 & -0.8013 \\ 0.9915 & 0.5844 \end{bmatrix}.$$

With STFT, the estimation of the mixing matrix $\mathbf{A}$ is obtained:

$$\hat{\mathbf{A}}_{SFFT} = \begin{bmatrix} 0.1270 & -0.7918 \\ 0.9917 & 0.6105 \end{bmatrix}.$$

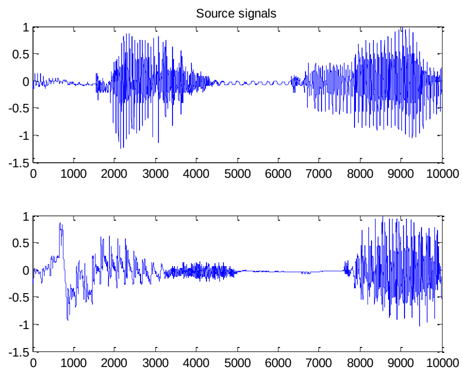
In this experiment, because $\mathbf{A}$ is invertible, we use $\hat{\mathbf{s}} = \hat{\mathbf{A}}^{-1}\mathbf{y} = \hat{\mathbf{A}}^{-1}\mathbf{A}\mathbf{s} \approx \mathbf{s}$ to obtain the estimation of source signals. Table 1 gives the distinction of STDCT and STFT from average NMSE, SNR and the computer time.

TABLE 1 The distinction of STDCT and STFT from average NMSE, SNR and average time

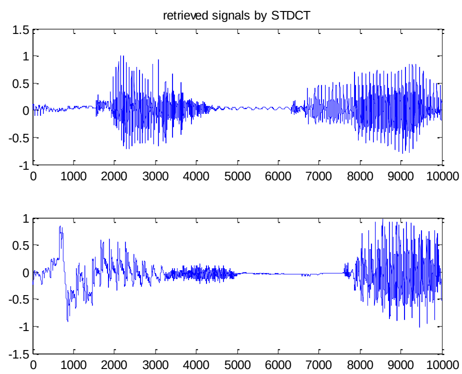
	Average MSE(dB)	Average SNR(dB)	Average time(s)
STDCT	40.0620	36.9546	5.816274
STFT	33.0461	29.6219	14.988418

From table 1, it is seen that the average NMSE and SNR with STDCT are better than these with STFT. Meanwhile, the average computer time with STDCT is less than STFT. The main reason is that the length of data with STFT is twice than the length of data with STDCT, it increases the runtime and decreases the accuracy of the cluster.

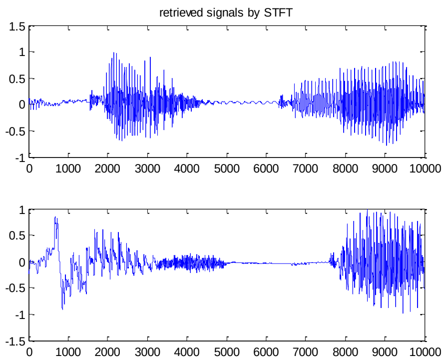
The retrieved source signals with STDCT are showed in Fig. 7(b) and the restored signals with STFT are showed in Fig. 7(c). The numerical values on the x-axis denote the discrete time sequence, while the numerical values on the y-axis represent amplitude of signals. Fig. 7(a) gives the source signals.



(a) Source signals



(b) Retrieved signals by STDCT



(c) Retrieved signals by STFT

FIGURE 7 The waves of the source signals and the restored signals with STDCT and STFT.

4.3 HIGH DIMENSION

In this experiment, $m=2$, $n=3$. The considered sources are from the experiment "six_src" in [15], each source has 65536 samples. The mixing matrix is followed by

$$A=\begin{bmatrix}0.7071 & 0.2588 & 0.9659\\ 0.7071 & -0.9659 & -0.2588\end{bmatrix}.$$

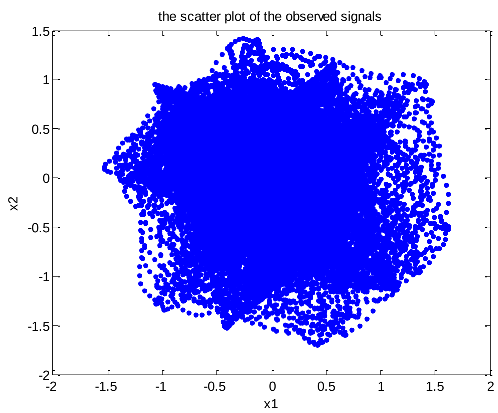
With STDCT, the estimation of the mixing matrix $\mathbf{A}$ is obtained:

$$\hat{A}_{STDCT}=\begin{bmatrix}0.7071 & -0.9658 & -0.2595\\ 0.7070 & 0.2587 & 0.9657\end{bmatrix}.$$

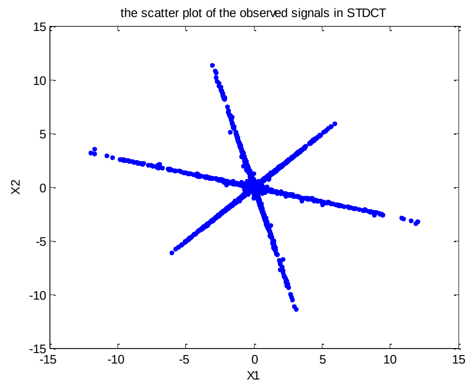
With STFT, the estimation of the mixing matrix $\mathbf{A}$ is obtained:

$$\hat{A}_{SFFT}=\begin{bmatrix}-0.2597 & -0.9652 & 0.7073\\ 0.9656 & 0.2605 & 0.7069\end{bmatrix}.$$

Fig.8 give the scatter plot $\mathbf{x}_1$ vs. $\mathbf{x}_2$ which mixed by three speech signals. Fig. 8(a) presents a scatter plot of the observed signals. Then the scatter plot of the STDCT domain data are shown in Fig. 8(b). In contrast, STDCT is obviously better than the scatter plot in time domain in this example.



(a) The observed signals $\mathbf{x}_1$ and $\mathbf{x}_2$



(b) The transformed domain of STDCT

FIGURE 8 Scatter plot $\mathbf{x}_1$ vs. $\mathbf{x}_2$ which mixed by three speech signals.

Almost all significant data are clustered along the three directions of the basis vectors. Because the data in STDCT have three clear directions, so we use K-means cluster directly, otherwise, we must adjust the cluster centers by using the improved method.

Table 2 shows the average NMSE and the average run-time for STDCT and STFT. From this table, it is seen that the NMSE with STDCT is better than STFT, at the same time, the average time with STDCT is less than STFT. Because in this experiment, the data with STDCT have clear directions, so we just repeat the step 4-6 once, then the runtime is less than Table 1.

TABLE 2 The distinction by using STDCT and STFT from average NMSE and average time

	Average NMSE(dB)	Average Time(s)
STDCT	67.2380	0.800644
STFT	58.3580	1.458367

5 Conclusions

In this paper, we proposed an improved K-means approach for solving the linear instantaneous mixtures. This method is based on assumption that there exist time domains (regions) or transformed domains (regions) where only one source occurs at a time. Unlike STFT, the STDCT transform real observed signals to real values without imaginary part, which decreases the computer time. Although we proposed an improved sparse BSS algorithm by using K-means clustering algorithm, other clustering approaches can instead of K-means clustering method to obtain similar results. The estimation error, computation time are reduced as compared to using STFT.

Acknowledgments

This work is supported by Natural Science Foundation of China (Grant No. 61102103 and Grant No. 61203287), Natural Science Foundation of Hubei Province (No. 2014CFB414) and the Special Fund for Basic Scientific Research of Central Colleges, China University of Geosciences Wuhan (Grand No. CUGL130247).

References

[1] P. Common, C. Jutten 2010 Handbook of blind source separation: independent component analysis and applications, Oxford: Elsevier.
[2] G. M. Zhang, Z. M. Cui, J.M. Chen, J. Wu 2010 CT image de-noising model based on independent component analysis and curvelet transform. Journal of Software, 5(9): 1006-1013.
[3] V. G. Reju, S. N. Koh, I.Y. Soon 2009 An algorithm for mixing matrix estimation in instantaneous blind source separation. Signal Processing. 89: 1762-1773.
[4] P. Bofill, M. Zibulevsky 2001 Underdetermined blind source separation using sparse representations. Signal Processing, 81(11): 2353-2362.
[5] Y. Li. A. Cichocki, S. Amari 2003 Sparse component analysis for blind source separation with less sensors than sources. In Proc 4th int Symp. Independent Component Analysis Blind Signal Separation. 89-94.
[6] Y. J. Zhang, H. M. Peng, H. G. Li 2012 Linear strength-based algorithm to sparse dependent sources separation. Journal of Information and Computational Science, 9(9): 2451-2464, 2012.
[7] Y. J. Zhang, H. M. Peng, H. W. Li. Wavelet packet transform-based algorithm for mixing matrix estimation. Journal of Computers. 7(11): 2605-2611.
[8] S. G. Kim, C. D. Yoo 2009 Underdetermined blind source separation based on subspace representation. IEEE Transactions on Signal Processing. 57(7): 2604-2614.
[9] Y. Deville, M. Puigt 2007 Temporal and time-frequency correlation-based blind source separation methods. Part I: Determined and underdetermined linear instantaneous mixtures. Signal Processing, 87: 374-407.
[10] C. C. Lin, P. F. Shui 2010 DCT-based reversible data hiding scheme. Journal of Software, 5(2): 214-224.
[11] Aissa-EI-Bey, N. Linh-Trung, K.Abed-Meraim 2007 Underdetermined blind separation of nondisjoint sources in the time-frequency domain. IEEE Transactions on Signal Processing. 55(3): 897-907.
[12] Y. Li, S. Amari, A. Cichocki 2007 Underdetermined blind source separation based on sparse representation. IEEE Transactions on Signal Processing, 55(8): 4004-4017.
[13] Z. Y. He, S. L. Xie, L. Zhang, A. Cichocki 2008 A note on Lewicki-Sejnowski gradient for learning overcomplete representation. Neural Computation, 20(3), 636-643.
[14] H. Hakan 2012 Discrete cosine transform tutorial. University of Houston. http://www.haberdar.org/ Discrete -Cosine-Transform-Tutorial. htm.
[15] Z. S. He, S. L. Xie and Y. L. Fu 2006 Sparse representation and blind source separation of ill-posed mixtures. Science In China Ser. E Information Sciences, 36(8): 864-879.
[16] Y. M. Li, B. Y. Ye 2011 Distance from a point to a line in n-dimensional space. Studies in college mathematics. 14(2): 3-5.

Authors

Yanni Zeng

Current position, grades: Ph. D. candidate of the Institute of Geophysics and Geomatics of China University of Geosciences, Wuhan, China.
University studies: B.E. degree in School of Mathematics and Computer from Hubei University, Wuhan, China, in 2002; M.S. degree in School of Mathematics and Statistics from Huazhong University of Science and Technology, Wuhan, China, in 2008.
Scientific interest: Blind source separation, array signal processing and estimating the number of signal source.
Experience: Teacher in institute of statistics of Hubei University of Economics, Wuhan, China.

Yujie Zhang, born in Hubei, China.

Current position, grades: lecturer at the China University of Geosciences, China.
University studies: M.S. degree in Applied Mathematics in 2006 from China University of Geosciences, China; PhD degree in the Institute of Geophysics and Geomatics in 2012 from China University of Geosciences, Wuhan, China.
Scientific interest: Blind Signal Processing, Time-frequency Analysis and their applications.

Rui Qi

Current position, grades: Ph. D. candidate of the Institute of Geophysics and Geomatics of China University of Geosciences, Wuhan, China.
University studies: B.E. degree in School of Mathematics and Physics from China University of Geosciences, Wuhan, China, in 2003; M.S. degree in School of Mathematics and Statistics from Huazhong University of Science and Technology, Wuhan, China, in 2009.
Scientific interest: Blind source separation, sparse representation and compressed sensing.
Experience: teacher at School of Science of Naval University of Engineering of China.