A geometric approach for separating several speech signals

Massoud Babaie-Zadeh[1,2], Ali Mansour[3], Christian Jutten[4], and Farrokh Marvasti[1,2] *

[1] Multimedia Lab, Iran Telecom Research Center (ITRC), Tehran, Iran
[2] Electrical Engineering Department, Sharif University of Technology, Tehran, Iran
[3] E3I2, ENSIETA, Brest, France
[4] Institut National Polytechnique de Grenoble (INPG), Laboratoire des Images et des Signaux (LIS), Grenoble, France
mbzadeh@yahoo.com, mansour@ieee.org
Christian.Jutten@inpg.fr, marvasti@itrc.ac.ir

Abstract. In this paper a new geometrical approach for separating speech signals is presented. This approach can be directly applied to separate more than two speech signals. It is based on clustering the observation points, and then fitting a line (hyper-plane) onto each cluster. The algorithm quality is shown to be improved by using DCT coefficients of speech signals, as opposed to using speech samples.

1 Introduction

Blind Source Separation (BSS) or Independent Component Analysis (ICA) consists in retrieving unknown statistically independent signals from their observed mixtures, assuming there is no information about the original source signals, or about the mixing system (hence the term *Blind*).

For linear instantaneous mixtures $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$, where the sources $\mathbf{s}(t) \triangleq (s_1(t), \dots, s_N(t))^T$ are (unknown) statistically independent signals, the observation signals are denoted $\mathbf{x}(t) \triangleq (x_1(t), \dots, x_N(t))^T$, and $\mathbf{A}$ is the $N \times N$ (unknown) mixing matrix. In this paper, the number of observations and sources are assumed to be equal. The problem is then to estimate the source vector $\mathbf{s}(t)$ only by knowing the observation vector $\mathbf{x}(t)$.

One approach to solve the problem is to determine a *separating matrix* $\mathbf{B}$ such that the outputs $\mathbf{y}(t) \triangleq \mathbf{B}\mathbf{x}(t)$ become statistically independent. This independence insures the estimation of the sources, up to a scale and a permutation indeterminacy [1].

Another approach is the geometric source separation algorithm, which has been first introduced in [2]. In this approach (for 2-dimensional case), it is first noted that because of the independence of source signals, $p_{s_1 s_2}(s_1, s_2) = p_{s_1}(s_1) p_{s_2}(s_2)$, where p stands for the Probability Density Function (PDF). Consequently, for bounded sources, the points (s_1, s_2) will be distributed in a rectangular region (Fig. 1-left). Now, because of the scale indeterminacy, the mixing

* This work has been partially funded by the European project Blind Source Separation and applications (BLISS, IST 1999-14190), by Iran Telecom Research Center (ITRC) and by Sharif university of technology.

C.G. Puntonet and A. Perieto (Eds.): ICA 2004, LNCS 3195, pp. 798-806, 2004

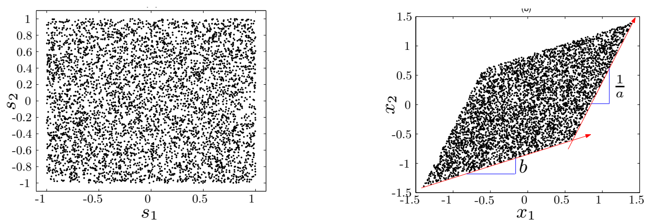

Fig. 1. Distribution of the sources (left) and observation (right) samples.

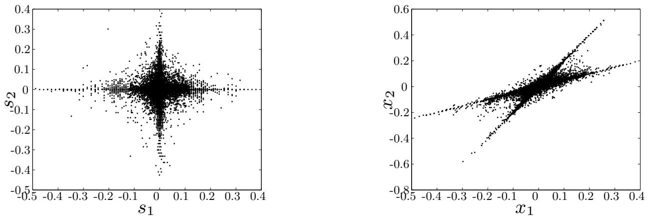

Fig. 2. Distribution of two speech samples (left), and their mixtures (right).

matrix is assumed to be of the form (normalized with respect to diagonal elements):

$$\mathbf{A} = \begin{pmatrix} 1 & a \\ b & 1 \end{pmatrix} \tag{1}$$

Under the transformation $\mathbf{x} = \mathbf{As}$, the rectangular region of the s-plane will be transformed into a parallelogram (Fig. 1-right), and the slopes of the borders of this parallelogram are $1/a$ and b. In other words, for estimating the mixing matrix, it is sufficient to determine the slopes of the borders of the distribution of the observation samples.

Although this approach is not easily generalized to higher dimensions, it is successful in separating two sources, provided that their distributions allow a good estimation of the borders of the parallelogram (*e.g.* uniform and sinusoidal sources). However, this technique cannot be used in separating speech signals because the PDF of a speech is mostly concentrated about zero. This comes from the fact that in a speech signal, there are many low energy (silence or unvoiced) sections. Consequently, as it can be seen in Fig. 2, it is practically impossible to find the borders of the parallelogram when the sources are speech signals. This is explained by a probabilistic manner in [3]: the probability of having a point in the borders of the parallelogram is very low.

Although for speech signals the borders of the parallelogram are not visible in Fig. 2, there are two visible "axes", corresponding to lines $s_1 = 0$ and $s_2 = 0$ in the s-plane (throughout the paper, it is assumed that the sources and hence the observations have zero-means). The slopes of these axes, too, determine a and b in (1). In other words, for speech signals, instead of finding the borders, we try to find these axes. This idea is used in [3] for separating speech signals by utilizing an "angular" histogram for estimating these axes. In this method, the resolution of the histogram cannot be too fine (requires more data points), and cannot be too coarse (bad estimation of the mixing matrix). Moreover, this approach cannot be easily generalized to mixtures of more than two speech signals.

In this paper, we propose another approach for estimating these "axes" based on line (or hyper-plane) fitting. The main idea is to fit two lines on the scatter plot of observations, which will be the required axes. This approach does not suffer from the problem of the resolution of a histogram. Moreover, we will see that this approach can be directly used in higher dimensions.

2 Speech separation by line fitting

2.1 Two dimensional case

As it is explained in the previous section, the main idea of our method is to estimate the slopes of two axes of the scatter plot of observations (Fig. 2-right). These axes corresponds to the lines $s_1 = 0$ and $s_2 = 0$ in the scatter plot of sources. The existence of these lines is a result of many low-energy sections of a speech signal. For example, the points with small s_1 and different values for s_2 will be concentrated about the axis $s_1 = 0$.

However, we do not use (1) as a model for mixing matrix, because it has two restrictions. Firstly, in this model, it is implicitly assumed that the diagonal elements of the actual mixing matrix are not zero, otherwise infinite values for a and b may be encountered (this situation corresponds to vertical axes in the x-plane). Secondly, this approach is not easy to be generalized to higher dimensions.

Instead of model (1), let us consider a general "separating matrix" $\mathbf{B} = [b_{ij}]_{2\times 2}$. Under the transformation $\mathbf{y} = \mathbf{B}\mathbf{x}$, one of the axes must be transformed to $y_1 = 0$, and the other to $y_2 = 0$. In other words, for every (x_1, x_2) on the first axis:

$$\begin{pmatrix} 0 \\ y_2 \end{pmatrix} = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \Rightarrow b_{11}x_1 + b_{12}x_2 = 0 \tag{2}$$

That is, the equation of the first axis is $b_{11}x_1 + b_{12}x_2 = 0$. In a similar manner, the second axis will be $b_{21}x_1 + b_{22}x_2 = 0$. Consequently, for estimating the separating matrix, the equations of the two axes must be found in the form of $\alpha_1 x_1 + \alpha_2 x_2 = 0$, and then each row of the separating matrix is composed of the coefficients of one of the axes. For finding the axes we suggest is to "fit" two straight lines on the scatter plot of the observations.

It is seen that by this approach, we are not restricted to non-vertical axes (non-zero diagonal elements of the mixing matrix). More interestingly, this approach can be directly used in higher dimensions, as stated below.

2.2 Higher dimensions

The approach stated above can be directly generalized to higher dimensions. For example, for 3 speech signals and 3 sources, the low-energy (silence and

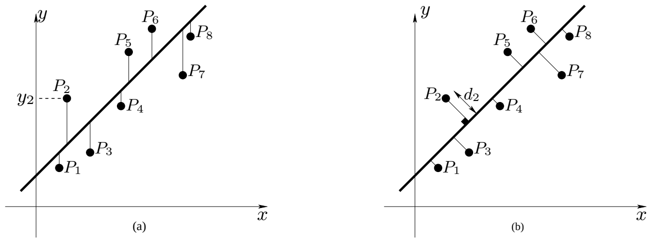

Fig. 3. Left) Least squares line fitting, Right) Orthogonal line fitting.

unvoiced) values of s_1 with different values of s_2 and s_3 will form the plane $s_1 = 0$ in the 3-dimensional scatter plot of sources. Hence, in this 3-dimensional scatter plot, there are 3 visible planes: $s_1 = 0$, $s_2 = 0$ and $s_3 = 0$. These planes will be transformed to three main planes in the scatter plot of observations. With calculations similar as (2), it is seen that each row of the separating matrix is composed of the coefficients of one of these main planes in the form of $\alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 = 0$.

Consequently, for N-dimensional case, N (hyper-)planes in the form of $\alpha_1 x_1 + \cdots + \alpha_N x_N = 0$ must be first "fitted" onto the scatter plot of observations. Then, each row of the separating matrix is the coefficients $(\alpha_1, \ldots, \alpha_N)$ of one of these (hyper-)planes.

3 Line fitting

To use the idea of the previous section, we need a method for fitting two lines (or N hyper-planes) onto the scatter plot of observations.

3.1 Fitting a straight line onto a set of points

First of all, consider the problem of fitting a line onto K data points $(x_i, y_i)^T$, $i = 1 \ldots K$. In the traditional least squares method, this is done by finding the line $y = mx + h$ which minimizes $\sum_{i=1}^{K} (y - y_i)^2 = \sum_{i=1}^{K} (mx_i + h - y_i)^2$. This is equivalent to minimizing the "vertical" distances between the line and the data points, as shown in left of Fig. 3. This technique is mainly used in linear regression analysis where there are errors in y_i's, but not in x_i's.

However, in our application of fitting a line onto a set of points, a better measure is minimizing the sum of "orthogonal distances" between the points and the line, as shown in the right part of Fig. 3. Moreover, as discussed in the previous sections, we are seeking a line in the form $ax + by = 0$. Consequently, the best fitted line is determined by minimizing $\sum_{i=1}^{K} d_i^2$, where d_i is the orthogonal distance between the i-th point and the line:

$$d_i = \frac{|ax_i + by_i|}{\sqrt{a^2 + b^2}} \tag{3}$$

However, $ax + by = 0$ is not uniquely determined by a pair (a, b), because (ka, kb) represents the same line. To obtain a unique solution, the coefficients are normalized such that $a^2 + b^2 = 1$. To summarize, the best fitted line $ax + by = 0$ is obtained by minimizing $\sum_{i=1}^{K}(ax_i + by_i)^2$ under the constraint $a^2 + b^2 = 1$.

N-Dimensional case. In a similar manner, an N-dimensional hyper-plane $\alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_N x_N = 0$ is fitted onto a set of K data points $\mathbf{x}_i = (x_1^{(i)}, x_2^{(i)}, \ldots, x_N^{(i)})^T$, $i = 1, \ldots, K$ by minimizing the cost function:

$$\mathcal{C}(\alpha_1, \ldots, \alpha_N) = \sum_{i=1}^{K} \left(\alpha_1 x_1^{(i)} + \cdots + \alpha_N x_N^{(i)} \right)^2 \tag{4}$$

under the constraint $g(\alpha_1, \ldots, \alpha_N) \equiv \alpha_1^2 + \cdots + \alpha_N^2 - 1 = 0$.

Solution. Using Lagrange multipliers, the optimum values for $\alpha_1, \ldots, \alpha_N$ satisfy $\nabla\mathcal{C} = \lambda \nabla g$. After a few algebraic calculations, this equation is written in the matrix form:

$$\mathbf{R}_{\mathbf{x}} \boldsymbol{\alpha} = \frac{\lambda}{K} \boldsymbol{\alpha} \tag{5}$$

where $\boldsymbol{\alpha} \triangleq (\alpha_1, \ldots, \alpha_N)^T$ and $\mathbf{R}_{\mathbf{x}} \triangleq \frac{1}{K} \sum_{i=1}^{K} \mathbf{x}_i \mathbf{x}_i^T$ is the correlation matrix of data points. Equation (5) shows that λ/K and $\boldsymbol{\alpha}$ are eigen value and eigen vector of the correlation matrix $\mathbf{R}_{\mathbf{x}}$, respectively. Moreover:

$$\mathcal{C} = \sum_{i=1}^{K} \left(\boldsymbol{\alpha}^T \mathbf{x}_i \right)^2 = \sum_{i=1}^{K} \boldsymbol{\alpha}^T \mathbf{x}_i \mathbf{x}_i^T \boldsymbol{\alpha} = K \boldsymbol{\alpha}^T \mathbf{R}_{\mathbf{x}} \boldsymbol{\alpha} = \lambda \boldsymbol{\alpha}^T \boldsymbol{\alpha} = \lambda$$

and hence for minimizing the cost function, λ must be minimum. Consequently, the solution of the hyper-plane fitting problem is given by the eigen vector of the correlation matrix which corresponds to its minimum eigen value.

Discussion. It is interesting to think about the conjunction of the above approach to Principal Component Analysis (PCA). Note that $\boldsymbol{\alpha}$ is the vector perpendicular to the plane $\alpha_1 x_1 + \cdots + \alpha_N x_N = 0$, and the above theorem states that this vector must be chosen in the direction with the minimum spread of data points, which is compatible with our heuristic interpretations of plane (line) fitting. This method has old foundations in mathematics [4], and somewhat called Principal Component Regression (PCR) [5].

3.2 Fitting 2 straight lines (N hyper-planes)

However, as stated in Section 2, for 2 mixtures of 2 sources our problem is to fit 2 lines onto the observation points, not just 1 line. In other words, as it is seen in Fig. 2, we need to divide the data points into 2 clusters, and then to fit a line onto the points of each cluster. The extension to N mixtures of N sources is straightforward: we need to divide the data into N clusters, and then to fit a hyper-plane onto the points of each cluster. Mathematically, this is equivalent to minimizing the following cost function (for the N-dimensional case):

$$\mathcal{C} = \sum_{\mathbf{x}_i \in \mathcal{S}_1} d^2(\mathbf{x}_i, l_1) + \sum_{\mathbf{x}_i \in \mathcal{S}_2} d^2(\mathbf{x}_i, l_2) + \cdots + \sum_{\mathbf{x}_i \in \mathcal{S}_N} d^2(\mathbf{x}_i, l_N) \tag{6}$$

- Initially distribute the points into clusters $\mathcal{S}_1, \ldots, \mathcal{S}_N$ (*e.g.* random initialization).
- Loop:
  1. Fit a line (hyper-plane) onto each set of points $\mathcal{S}_i$ (we call it l_i).
  2. Recalculate the clusters: Let $\mathcal{S}_i$ be the set of all points which are closer to line (hyper-plane) l_i than other lines (hyper-planes), that is:

$$\mathcal{S}_i = \{\mathbf{x} \mid d(\mathbf{x}, l_i) < d(\mathbf{x}, l_j), \forall j \neq i\}$$

- Repeat until convergence.

Fig. 4. Algorithm of fitting two lines (N hyper-planes) onto a set of points.

where $\mathcal{S}_j$ is the j-th cluster of points and $d^2(\mathbf{x}_i, l_j)$ denotes the perpendicular distance of the i-th point from the j-th plane.

Having divided the points into clusters $\mathcal{S}_1, \ldots, \mathcal{S}_N$, the previous section gives us the best line fitted onto the points of each cluster. For clustering the data points, we use the algorithm stated in Fig. 4, which is inspired from the k-means (or Lloyd) algorithm for data clustering [6]. Its difference with k-means is that in k-means, each cluster is mapped onto a point (point $\rightarrow$ point), but in our algorithm each cluster is mapped onto a line or hyper-plane (point $\rightarrow$ line).

The following theorem is similar to a corresponding theorem for the k-means algorithm [6].

Theorem 1. *The algorithm of Fig. 4 converges in a finite number of iterations.*

Proof. At each iteration, the cost function (6) cannot be increased. This is because in the first step (fitting hyper-planes onto the clusters) the cost function is either decreased or does not change. In the second step, too, the redistribution of the points in the clusters is done such that it decreases the cost function or does not change it. Moreover, there is a finite number of possible clustering of finite number of points. Consequently, the algorithm must converge in a finite number of iterations. □

Initialization. The fact that the cost-function is non-increasing in the algorithm, shows that the algorithm may get trapped in a local minimum. This is one of major problems of the k-means algorithm, too. It depends on the initialization of the algorithm, and become more severe when the dimensionality increases. In k-means, one approach is to run the algorithm with several randomly chosen initializations, and then to take the result which produces the minimum cost-function.

4 Final algorithm, and its improvement by using DCT

The final separation algorithm is now evident. First, run the algorithm of Fig. 4. After convergence, there are N lines (hyper-planes) $l_i : \alpha_{i1}x_1 + \cdots + \alpha_{iN}x_N = 0$, $i = 1 \ldots N$. Then, the i-th row of the separating matrix is $(\alpha_{i1}, \ldots, \alpha_{iN})$.

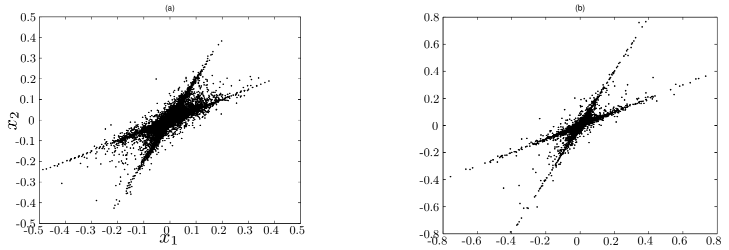

Fig. 5. Distribution of the observations (left) and their DCT coefficients (right).

However, the separation quality of the algorithm can be improved, with a simple trick. Recall that the success of the algorithm is because of the existence of two visible "axes" in Fig. 2. These axes were formed because of the small-valued (low-energy) parts of one speech and other parts of the second one. Now, recall that the Discrete Cosine Transform (DCT) coefficients of a speech frame (10-20 msec) contain a lot of nearly zero values. Moreover, DCT is a linear transformation, and hence, the DCT coefficients of the observations are a mixture of the DCT coefficients of the original speeches with the same mixing matrix. Therefore, it seems that it is a good idea to apply the algorithm on the DCT coefficients of observations instead of themselves. Figure 5 shows an example of the scatter plot of observations, and that of their DCT coefficients. It is seen visually that the "axes" are more visible in the scatter plot of DCT coefficients. Consequently, one expects to get better results by applying the algorithm on the DCT coefficients of the observations, as is confirmed by our experiments, too.

5 Experimental Results

The algorithm has been tested on several speech signals and the results for 4 typical speech signals are presented here. Let these typical speech signals be called s_1 through s_4. Each of them has 20000 samples (with 8KHz sampling rate). In all the experiments, the diagonal elements of the mixing matrix are 1, while all other elements are 0.5. The initialization of the algorithm is done randomly.

To measure the performance of the algorithm, let $\mathbf{C} \triangleq \mathbf{BA}$ be the global mixing-separating matrix. Then, our performance criterion is the Signal to Noise Ratio defined by (assuming no permutation):

$$\mathrm{SNR}_i(\text{in dB}) \triangleq 10 \log_{10} \frac{c_{ii}^2}{\sum_{j \neq i} c_{ij}^2} \tag{7}$$

This criterion shows how much the global matrix $\mathbf{C}$ is close to the identity matrix (it is the real SNR provided that the sources have equal energies). The results of applying the algorithm for separating 2, 3 and 4 mixed speech signals are given in the Table 1. It must be noted that the SNR values are typical, they depend on the initialization. This table clearly shows the ability of the algorithm, and the advantage of using DCT coefficients (specially in lower dimensions).

In our experiments, for 2-dimensional case, local minima have never been encountered. For 3-dimensional case, they were encountered from time to time. In 4-dimensional case, they were frequently encountered, specially when we have not used the DCT coefficients. This fact shows the necessity of finding cleverer initialization approaches for higher dimensions.

Table 1. Separation results for 4 typical speech signals

Signals Mixed	Without using DCT				Number of iterations	With DCT			
	SNR_1	SNR_2	SNR_3	SNR_4		SNR_1	SNR_2	SNR_3	SNR_4
s_1, s_2	28dB	28dB	–	–	$\simeq 10$	35dB	35dB	–	–
s_3, s_4	26dB	25dB	–	–	$\simeq 10$	30dB	30dB	–	–
s_1, s_2, s_3	28dB	30dB	34dB	–	$\simeq 20$	34dB	37dB	39dB	–
s_2, s_3, s_4	25dB	28dB	29dB	–	$\simeq 20$	36dB	28dB	25dB	–
s_1, s_2, s_3, s_4	42dB	30dB	43dB	39dB	$\simeq 30$	44dB	30dB	44dB	33dB

6 Conclusion

In this paper, a geometrical approach for separating several speech signals has been presented. It has been shown that for speech signals (or other sources whose PDF's are concentrated about zero), the ICA can be accomplished by a clustering of observation samples and then applying a PCA on each cluster and taking the smallest principal component. Although this approach was based on geometric interpretations, its final algorithm is completely algebraic.

Initialization is the main problem of this algorithm. Finding better initialization approaches is currently under study.

References

1. P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.
2. C. Puntonet, A. Mansour, and C. Jutten, "A geometrical algorithm for blind separation of sources," in *Actes du XVème Colloque GRETSI 95*, Juan-Les-Pins, France, Septembre 1995, pp. 273–276.
3. A. Prieto, B. Prieto, C. G. Puntonet, A. Cañas, and P. Martín-Smith, "Geometric separation of linear mixtures of sources: Application to speech signals," in *ICA99*, Aussois, France, January 1999, pp. 295–300.
4. K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, vol. 2, pp. 559–572, 1901.
5. W. F. Massy, "Principal component regression in exploratory statistical research," *Journal of American Statistical Association*, vol. 60, pp. 234–256, March 1965.
6. A. Gersho and R. M. Gray, *Vector Quantization and signal compression*, Kluwer Academic Publishers, 1992.