# BLIND SOURCE SEPARATION VIA MULTINODE SPARSE REPRESENTATION

**Michael Zibulevsky**
Department of Electrical Engineering
Technion, Haifa 32000, Israel
*mzib@ee.technion.ac.il*

**Pavel Kisilev**
Department of Electrical Engineering
Technion, Haifa 32000, Israel
*paulk@tx.technion.ac.il*

**Yehoshua Y. Zeevi**
Department of Electrical Engineering
Technion, Haifa 32000, Israel
*zeevi@ee.technion.ac.il*

**Barak Pearlmutter**
Department of Computer Science
University of New Mexico
Albuquerque, NM 87131 USA
*bap@cs.unm.edu*

## Abstract

The blind source separation problem is concerned with extraction of the underlying source signals from a set of their linear mixtures, where the mixing matrix is unknown. It was discovered recently, that exploiting the sparsity of sources in an appropriate representation according to some signal dictionary, dramatically improves the quality of separation. In this work we use the property of multiscale transforms, such as wavelet or wavelet packets, to decompose signals into sets of local features with various degrees of sparsity. We use this intrinsic property for selecting the best (most sparse) subsets of features for further separation. The performance of the algorithm is verified on noise-free and noisy data. Experiments with simulated signals, musical sounds and images demonstrate significant improvement of separation quality over previously reported results.

## 1 Introduction

In the blind source separation problem an $N$-channel sensor signal $\mathbf{x}(\xi)$ is generated by $M$ unknown scalar source signals $s_m(\xi)$, linearly mixed together by an unknown $N \times M$ mixing, or crosstalk, matrix $\mathbf{A}$, and possibly corrupted by additive noise $\mathbf{n}(\xi)$:

$$\mathbf{x}(\xi) = \mathbf{A}\mathbf{s}(\xi) + \mathbf{n}(\xi).$$

The independent variable $\xi$ is either time or spatial coordinates in the case of images. We wish to estimate the mixing matrix $\mathbf{A}$ and the $M$-dimensional source signal $\mathbf{s}(\xi)$.

A classical example of blind source separation is the so-called cocktail party problem, wherein it is desirable to separate several speakers from their audio recorded mixtures. One promising application in 2D is encountered in hyperspectral imaging, wherein images of a body surface are taken at several wavelengths. If several chemical compounds are present

on a surface, the image at each wavelength represents a weighted sum of fingerprints of the unknown concentrations of the various compounds, with weights determined by radiation spectra of each compound. The problem is to recover unknown concentrations and spectra.

The assumption of statistical independence of the source components $s_m(\xi)$ leads to the Independent Component Analysis (ICA) [1, 2]. A stronger assumption is the sparsity of decomposition coefficients, when the sources are properly represented [3]. In particular, let each $s_m(\xi)$ have a sparse representation obtained by means of its decomposition coefficients $c_{ml}$ according to a signal dictionary of functions $\varphi_l(\xi)$:

$$s_m(\xi) = \sum_l c_{ml}\, \varphi_l(\xi). \tag{1}$$

The functions $\varphi_l(\xi)$ are called *atoms* or *elements* of the dictionary. These elements do not have to be linearly independent, and instead may form an overcomplete dictionary,e.g. wavelet-related dictionaries (wavelet packets, stationary wavelets, *etc.*, see for example [4] and references therein). Sparsity means that only a small number of coefficients $c_{ml}$ differ significantly from zero. Then, unmixing of the sources is performed in the transform domain, i.e. in the domain of these coefficients $c_{ml}$. The property of sparsity often yields much better source separation than standard ICA techniques, and can work well even with more sources than mixtures. In many cases there are distinct groups of coefficients,wherein sources have different sparsity properties. The key idea in this study is to select only a subset of features (coefficients) which is best suited for separation, with respect to the following criteria: (1) sparsity of coefficients (2) separability of sources' features. After this subset is formed, one uses it in the separation process, which can be accomplished by standard ICA algorithms or by clustering. The performance of our algorithm is verified on noise-free and noisy data. Our experiments with 1D signals and images demonstrate that the proposed method further improves separation quality, as compared with result obtained by using sparsity of all decomposition coefficients.

## 2 Motivating example: sparsity of random blocks in the Haar basis

To provide intuitive insight into the practical implications of our main idea, we first use 1D block functions, that are piecewise constant, with random amplitude and duration of each constant piece (Figure 1). Since images are 2D piece-wise smooth functions, the implications are similar in the 2D case.

It is known, that the Haar wavelet basis provides compact representation of such functions. Let us take a close look at the Haar wavelet coefficients at different resolution levels *j=0,1,...,J*. Wavelet basis functions at the finest resolution level *j=J* are obtained by translation of the Haar mother wavelet: $\varphi(t) = \{1, \text{if } t \in [0,1);\ -1, \text{if } t \in [1,2);\ 0 \text{ otherwise}\}$. Taking the scalar product of a function $s(t)$ with the wavelet $\varphi_J(t-\tau)$, we produce a finite differentiation of the function $s(t)$ at the point $t = \tau$. This means that the number of non-zero coefficients at the finest resolution for a block function will correspond roughly to the number of jumps of this function. Proceeding to the next, coarser resolution level, we have $\varphi_{J-1}(t) = \{1, \text{if } t \in [0,2);\ -1, \text{if } t \in [2,4);\ 0 \text{ otherwise}\}$. At this level, the number of non-zero coefficients still corresponds to the number of jumps, but the total number of coefficients at this level is halved , and so is the sparsity. If we further proceed to coarser resolutions, we will encounter levels where the support of a wavelet $\varphi_j(t)$ is comparable to the typical distance between jumps in the function $s(t)$. In this case, most of the coefficients are expected to be nonzero, and, therefore, sparsity will fade away.

To demonstrate how this influences accuracy of a blind source separation, we randomly generated two block-signal sources (Figure 1, two upper plots.), and mixed them by the

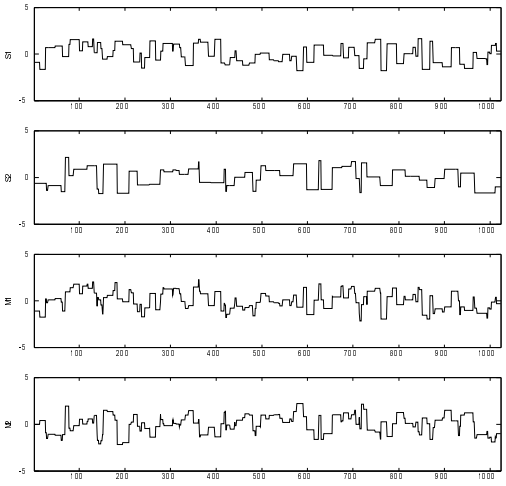


Figure 1: Random block signals (two upper) and their mixtures (two lower)

cross talk matrix

$$\mathbf{A} = \begin{pmatrix} 0.8321 & 0.6247 \\ -0.5547 & 0.7809 \end{pmatrix}.$$

Resulting sensor signals, or mixtures, $x_1(t)$ and $x_2(t)$ are shown in the two lower plots of Figure 1. The scatter plot of $x_1(t)$ versus $x_2(t)$ does not exhibit any visible distinct orientations (Figure 2, left). Similarly, in the scatter plot of the wavelet coefficients at the lowest resolution distinct orientations are hardly detectable (Figure 2, middle). In contrast, the scatter plot of the wavelet coefficients at the highest resolution (Figure 2, right) depicts two distinct orientations, which correspond to the columns of the mixing matrix. Indeed, if only one source, say $s_1(t)$, was present, the sensor signals would look like

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) \\ x_2(t) &= a_{21}s_1(t) \end{aligned}$$

and the points at the scatter plot of $x_2$ versus $x_1$ would belong to the straight line placed along the vector $[a_{11}a_{21}]^T$. The same thing happens, when two sources are present, but the coefficients are sparse: at each particular index, where a coefficient of the first source is large, there is a high probability, that the corresponding coefficient of the second source is small, and the point at the scatter plot still lies close to the mentioned straight line. The same arguments are valid for the second source. This explains the emergence of two dominant orientations at the scatter plot.

## 3 Multinode sparse source separation

### 3.1 Two approaches to sparse source separation: Infomax and Clustering

There are several ways to separate sparse sources. As was pointed in the context of sparse raw signals and in the context of sparse coefficients [3], the Bell-Sejnowski (BS) InfoMax objective [1]

$$\min_W \; -K \log |\det \mathbf{W}| + \sum_{n=1}^{N} \sum_{k=1}^{K} \nu((\mathbf{WY})_{nk}) \tag{2}$$

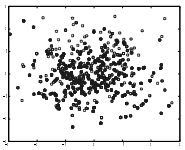
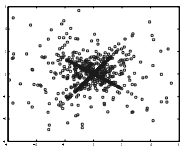
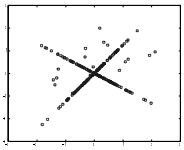
| | Raw signals | All wavelet coefficients | High resolution WT coefficients |
|---|---|---|---|
| |  |  |  |
| InfoMax | 13.9 | 4.2 | 0.69 |
| FCM | 13.3 | 2.4 | 0.41 |

Figure 2: Separation of block signals: scatter plots of sensor signals (left), and of their wavelet coefficients (middle and right). Lower columns present the normalized mean-squared separation error (%) corresponding to the Bell-Sejnowski InfoMax, and to the Fuzzy C-Means clustering, respectively.

is suitable for the case of equal number of sources and sensors, when the scalar function $\nu(\cdot)$ is a smooth approximation of an absolute value function. Here $\mathbf{W} = \mathbf{A}^{-1}$ is the *unmixing* matrix, to be estimated, and $\mathbf{Y}$ is the features', or (new) data, matrix whose rows are either the samples of sensor signals or their decomposition coefficients, $N$ is the number of sensors, and $K$ is the number of features, or coefficients.

Another approach to the separation of sparse sources is clustering along orientations of data concentration in the scatter plot. This works efficiently even if the number of sources is greater than the number of sensors. In order to determine orientations of scattered data, we project the data points onto the surface of a unit sphere by normalizing corresponding vectors, and then apply a standard clustering algorithm.

Our *clustering procedure* can be summarized as follows:

1. Form the feature matrix $\mathbf{Y}$, by putting samples of the sensor signals or (*subset of*) their decomposition coefficients into the corresponding rows of the matrix;

Each column $\mathbf{y}_k$ of the matrix $\mathbf{Y}$ represents a data point in an $N$-dimensional space.

2. Normalize feature vectors: $\mathbf{y}_k = \mathbf{y}_k / \|\mathbf{y}_k\|_2$, in order to project data points onto the surface of a unit sphere, where $\|\cdot\|_2$ denotes the $l_2$ norm;

Before normalization, it is reasonable to remove data points with a very small norm, since these very likely are noisy.

3. Move data points to a half-sphere, e.g. by forcing the sign of the first coordinate $y_k^1$ to be positive: IF $y_k^1 < 0$ THEN $\mathbf{y}_k = -\mathbf{y}_k$;

Without this operation each set of linearly (i.e., along a line) clustered data points would yield two clusters on opposite sides of the sphere.

4. Estimate cluster centers by using some clustering algorithm. The coordinates of these centers will form the columns of the estimated mixing matrix $\tilde{\mathbf{A}}$;

We used *Fuzzy C-means* (FCM) clustering algorithm as implemented in Matlab Fuzzy Logic Toolbox.

5. Estimate the sources: $\tilde{\mathbf{s}}(t) = \tilde{\mathbf{A}}^{-1}\mathbf{x}(t)$.

Note that the estimated unmixing matrix $\tilde{\mathbf{A}}^{-1}$ obtained by using the new feature set, is applied to the original sensor signals in order to recover sources in their original domain.

The above clustering operation is applied to various feature sets. We should stress here

that our method is *not* restricted to estimation of *square* mixing matrices, although the estimation of sources (step 5 in the above algorithm) is more complicated in the rectangular cases.

### 3.2 Results of separation of random blocks using the Haar wavelet basis.

In order to measure the separation accuracy, we normalize the original sources $s_m(t)$ and the estimated sources $\tilde{s}_m(t)$. The (normalized) squared error (NSE) is then computed as $\|\tilde{s}_m - s_m\|_2 / \|s_m\|_2$. Resulting separation errors for block sources are presented in the lower part of Figure 2. The largest error (13%) are obtained on the raw data, and the smallest (below 0.7%) – on the wavelet coefficients at the highest resolution, which have the best sparsity. Using all wavelet coefficients yields intermediate sparsity and performance.

### 3.3 Adaptive selection of sparse subsets of coefficients in wavelet packet tree.

Our choice of a particular wavelet basis and of the sparsest subset of coefficients was obvious in the above example: it was based on knowledge of the structure of piecewise constant signals. For sources having oscillatory components (like sounds or images with textures), other systems of basis functions, for example, wavelet packets, might be more appropriate. The wavelet packet library consists of the triple-indexed family of functions:

$$\varphi_{j,i,q}(t) = 2^{j/2}\varphi_q(2^j t - i),\ j, i \in \mathbf{Z},\ q \in \mathbf{N}. \tag{3}$$

where $j, i$ are the scale and shift parameters, respectively, and $q$ is the frequency parameter. [Roughly speaking, $q$ is proportional to the number of oscillations of a mother wavelet $\varphi_q(t)$]. These functions form a binary tree whose nodes are indexed by two indices: the depth of the level $j$ and the number of node $q = 0, 1, 2, 3, ..., 2^j$–1 at the specified level $j$.

When signals have a complex nature, it is difficult to decide in advance which nodes contain the sparsest sets of coefficients. That is why we use the following simple *adaptive approach*. First, for every node of the tree, we apply our clustering algorithm, and compute a measure of clusters' distortion. In our experiments we used a standard *global distortion*: the mean squared distance of data points to the centers of their own (closest) clusters. Second, we choose a few best nodes with the minimal distortion, combine their coefficients into one data set, and apply a separation algorithm (clustering or Infomax) to these data.

## 4 Experimental results

The proposed blind separation method based on the wavelet-packet approach, was evaluated by using several types of signals. We have already discussed the relatively simple example of a random block signal. The second type of signal is a frequency modulated (FM) sinusoidal signal. The carrier frequency is modulated by either a sinusoidal function (FM signal) or by random blocks (BFM signal). The third type is a musical recording of flute sounds. Finally, we apply our algorithm to images. An example of such images is presented in the left part of Figure 3. Source images and their mixtures are shown at the upper two sets of plots, and the estimated images are shown in the lower two plots.

In order to compare accuracy of our method with that attainable by other methods, we form the following feature sets: (1) raw data, (2) Short Time Fourier Transform (STFT) coefficients for 1D signals, and Discrete Cosine Transform (DCT) coefficients for images, (3) Wavelet packet coefficients at the 'best' nodes, using various mother wavelets.

On the right part of Figure 3 we show an example of scatter plots of the wavelet packet coefficients obtained at different nodes of the wavelet packet tree. The upper left scatter plot, marked with 'C', corresponds to the complete set of coefficients at all nodes. The

rest are the scatter plots of sets of coefficients indexed on a wavelet packet tree. Generally speaking, the more distinct the two dominant orientations appear on these plots, the more precise is the estimation of the mixing matrix, and, therefore, the better is the quality of separation. Note, that only two nodes, $c_{22}$ and $c_{23}$, show clear orientations. These nodes will most likely be selected by the algorithm for further estimation process.

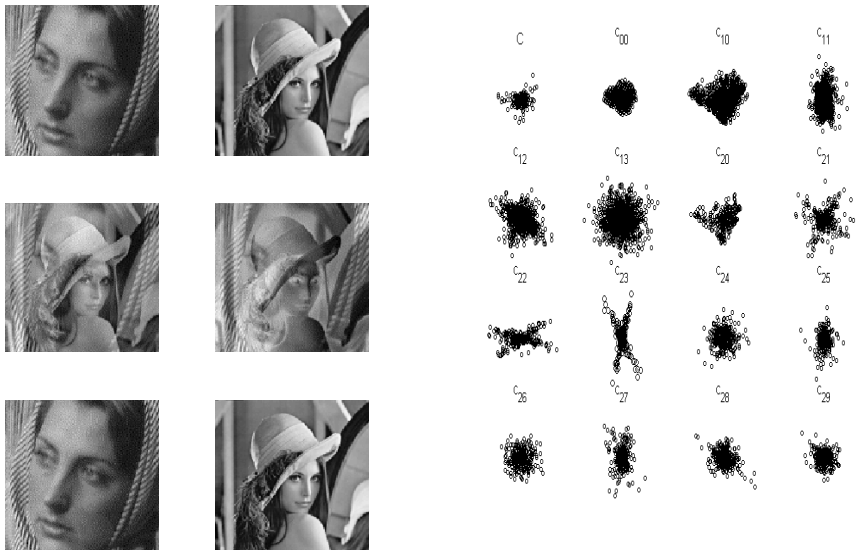



Figure 3: Left: two source images (upper pair), their mixtures (middle pair) and estimated images (lower pair). Right: scatter plots of the wavelet packet (WP) coefficients of mixtures of images; subsets are indexed on the WP tree.

| 1D Signals | raw data | STFT | WP |
|---|---|---|---|
| Blocks | 31.89 | 16.31 | 0.43 |
| BFM | 49.81 | 8.17 | 4.48 |
| FM | 50.57 | 5.66 | 4.13 |
| Flutes | 12.18 | 5.36 | 3.93 |
| | raw data | DCT | WP |
| Images | 22.11 | 19.11 | 6.04 |

Table 1: Experimental results: normalized mean-squared separation error (%) for *noise-free* signals and images, applying the FCM separation to raw data and decomposition coefficients in various domains. In the case of wavelet packets (WP) the best nodes selected by our algorithm were used.

| SNR [dB] | $\infty$ | 12 | 11 | 10 | 8 |
|---|---|---|---|---|---|
| Mixtures of images w. white gaussian noise | 2.05 | 4.38 | 7.12 | 12.76 | 41.70 |
| Mixtures of images w. salt&pepper noise | 2.05 | 2.17 | 2.93 | 4.90 | 14.61 |

Table 2: Performance of the algorithm in presence of various sources of noise in mixtures: normalized mean-squared separation error (%) for images, applying our adaptive approach along with the BS InfoMax separation.

Table 1 summarizes results of experiments in which we applied our algorithm along with the FCM separation to each noise-free feature set and compared normalized mean-squared errors (NMSE). From Table 1 it is clear that using our adaptive 'best' nodes method outperforms all other feature sets for each type of signal. Similar improvement was achieved by using our algorithm along with the BS InfoMax separation, which provided even better results for images.

In order to verify the performance of our method in presence of noise, we added various noise (white gaussian and salt&pepper) to mixtures of images at various signal-to-noise ratios (SNR). Table 2 summarizes these experiments in which we applied our algorithm along with the BS InfoMax separation. Our algorithm provides reasonable separation quality for SNR's of about 10 dB and higher in the case of salt&pepper noise, and for SNR's of about 11 dB and higher in the case of white gaussian noise. More experimental results, as well as parameters of simulations, can be found in [7].

## 5 Conclusions

Experiments with both one- and two-dimensional simulated and natural signals demonstrate that sparse representations improve the efficiency of blind source separation. The proposed method improves the separation quality by utilizing the structure of signals, wherein several subsets of the wavelet packet coefficients have significantly better sparsity and separability than others. In this case, scatter plots of these coefficients show distinct orientations each of which specifies a column of the mixing matrix. Further, projecting points appearing on the scatter plot onto the surface of a unit sphere, facilitates the separation into distinct data clusters. We choose the 'good subsets' according to the global distortion adopted as a measure of cluster quality. Finally, we combine together coefficients from the best chosen subsets and restore the mixing matrix using only this new subset of coefficients by the Infomax algorithm or clustering. This yields significantly better experimental results than those obtained by using standard Infomax and clustering approaches.

#### Acknowledgements

Research supported in part by the Ollendorff Minerva Center, by the Fund for Promotion of Research at the Technion and by the Israeli Ministry of Science.

## References

[1] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[2] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, no. 2, pp. 94–128, 1999.

[3] M. Zibulevsky and B. A. Pearlmutter, "Blind separation of sources with sparse representations in a given signal dictionary," *Neural Computation*, vol. 13, no. 4, pp. 863–882, 2001.

[4] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[5] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[6] *International Workshop on Independent Component Analysis and Blind Signal Separation*, (Helsinki, Finland), June 19–20 2000. In press.

[7] P. Kisilev, M. Zibulevsky, Y. Zeevi, and B. Pearlmutter, *Multiresolution framework for sparse blind source separation*, CCIT Report no.317, June 2000