

# BLIND SOURCE SEPARATION VIA MULTINODE SPARSE REPRESENTATION

Pavel Kisilev\*, Michael Zibulevsky\*, Yehoshua Y. Zeevi\*, and Barak A. Pearlmutter#

\*Dept. of Electrical Engineering, Technion, Israel Institute of Technology, Haifa 32000, Israel

#Department of Computer Science, University of New Mexico, Albuquerque, NM 87131 USA

## 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  $\mathbf{n}(\xi)$ :

$$\mathbf{x}(\xi) = \mathbf{A}\mathbf{s}(\xi) + \mathbf{n}(\xi). \quad (1)$$

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  $s_m(\xi)$ ,  $m = 1, \dots, M$  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  $c_{mk}$  according to a signal dictionary of functions  $\varphi_k(\xi)$ :

$$s_m(\xi) = \sum_k c_{mk} \varphi_k(\xi). \quad (2)$$

The functions  $\varphi_k(\xi)$  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 [10] and references therein). Sparsity means that only a small number of coefficients  $c_{mk}$  differ significantly from zero. Then, unmixing of the sources is performed in the transform domain, i.e. in the domain of these coefficients  $c_{mk}$ . The property of sparsity often yields much better source separation than standard ICA, 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.

---

Supported in part by the Ollendorff Minerva Center, by the Israeli Ministry of Science, by NSF CAREER award 97-02-311 and by the National Foundation for Functional Brain Imaging

## 2. TWO APPROACHES TO SPARSE SOURCE SEPARATION: INFOMAX AND CLUSTERING

### 2.1. InfoMax

Sparse sources can be separated by each one of several techniques. For example, the Bell-Sejnowski Information Maximization (BS InfoMax) approach [1], which, under the assumption of a noiseless system and a square mixing matrix in (1), is equivalent to the maximum likelihood (ML) formulation of the problem [4], [5], can be applied.

For the sake of simplicity of the presentation, let us consider the case where the dictionary of functions used in a source decomposition (2) is an orthonormal basis. (In this case, the corresponding coefficients  $c_{mk} = \langle s_m, \varphi_k \rangle$ , where  $\langle \cdot, \cdot \rangle$  denotes the inner product). From (1) and (2) the decomposition coefficients of the noiseless mixtures, according to the same signal dictionary of functions  $\varphi_k(\xi)$ , are:

$$\lambda_k = \mathbf{A} \mathbf{c}_k, \quad (3)$$

where  $M$ -dimensional vector  $\mathbf{c}_k$  forms the  $k$ -th column of the matrix  $\mathbf{C} = \{c_{mk}\}$ .

Let  $\mathbf{Y}$  be the *features*, or (new) data, matrix of dimension  $M \times K$ , where  $K$  is the number of features. Its rows are either the samples of sensor signals (mixtures), or their decomposition coefficients. In the last case, the coefficients  $\lambda_k$ 's form the columns  $\mathbf{y}_k$ 's of the matrix  $\mathbf{Y}$ . (In the following discussion we assume this setting, if not stated other). We are interested in the maximum likelihood estimate of  $\mathbf{A}$  given the data  $\mathbf{Y}$ .

Let the corresponding coefficients  $c_{mk}$  be independent random variables with a probability density function (pdf) of an exponential type

$$p_m(c_{mk}) \propto \exp\{-\nu(c_{mk})\}, \quad (4)$$

where the scalar function  $\nu(\cdot)$  is a smooth approximation of an absolute value function. Such kind of distribution is widely used for modeling sparsity [6], [7]. In view of the independence of  $c_{mk}$ , and (4), the prior pdf of  $\mathbf{C}$  is

$$p(\mathbf{C}) \propto \prod_{m,k} \exp\{-\nu(c_{mk})\}. \quad (5)$$

Taking into account the linear transformation (3), the parametric model for the pdf of  $\mathbf{Y}$  with respect to parameters  $\mathbf{A}$  is

$$p_{\mathbf{A}}(\mathbf{Y}) = \frac{p(\mathbf{C})}{|\det \mathbf{A}|^K}. \quad (6)$$

Let  $\mathbf{W} = \mathbf{A}^{-1}$  be the *unmixing* matrix, to be estimated. Then, substituting  $\mathbf{C} = \mathbf{W}\mathbf{Y}$ , combining (6) with (5) and

taking the logarithm we arrive at the log-likelihood function:

$$L_{\mathbf{W}}(\mathbf{Y}) = K \log |\det \mathbf{W}| - \sum_{m=1}^M \sum_{k=1}^K \nu((\mathbf{W}\mathbf{Y})_{mk}). \quad (7)$$

Maximization of  $L_{\mathbf{W}}(\mathbf{Y})$  with respect to  $\mathbf{W}$  is equivalent to the BS InfoMax, and can be solved efficiently by the Natural Gradient algorithm [8]. We used this algorithm as implemented in the ICA/EEG Matlab toolbox [9].

### 2.2. Clustering

Another approach to the separation of sparse sources is clustering along orientations of data concentration in the  $N$ -dimensional space wherein each column  $\mathbf{y}_k$  of the matrix  $\mathbf{Y}$  represents a data point. Let us consider a two-dimensional noiseless case, wherein two source signals,  $s_1(t)$  and  $s_2(t)$ , are mixed by a  $2 \times 2$  matrix  $\mathbf{A}$ , arriving at two mixtures  $x_1(t)$  and  $x_2(t)$ . Further, let the data matrix  $\mathbf{Y}$  be constructed from these mixtures  $x_1(t)$  and  $x_2(t)$ . If only one source, say  $s_1(t)$ , was present, the sensor signals would be

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

and the data points at the scatter diagram 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 *sparse* sources are present. In this sparse case, at each particular index where a sample of the first source is large, there is a high probability, that the corresponding sample of the second source is small, and the point at the scatter diagram still lies close to the mentioned straight line. The same arguments are valid for the second source. As a result, data points are concentrated around two dominant orientations (see for example the right scatter plot in Figure 2), which are directly related to the columns of  $\mathbf{A}$ .

Source signals are rarely sparse in their original domain. In contrast, their decomposition coefficients (2) usually are sparse. Therefore, we construct the data matrix  $\mathbf{Y}$  from the decomposition coefficients of mixtures (3), rather than from the mixtures themselves, and the above discussion is valid.

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. This clustering approach works efficiently even if the number of sources is greater than the number of sensors.

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;

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  $\hat{\mathbf{A}}$ ;

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

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

Note that the estimated unmixing matrix  $\hat{\mathbf{A}}^{-1}$  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. MULTINODE SOURCE SEPARATION

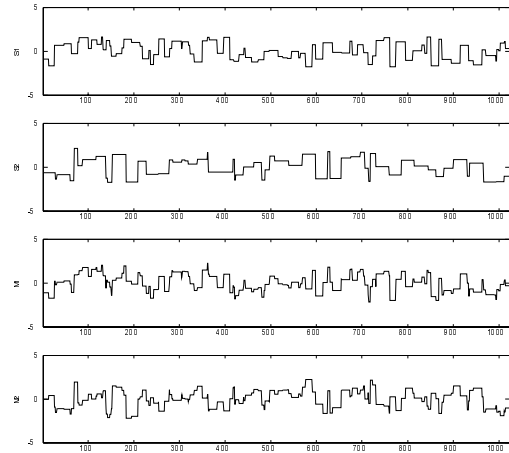
#### 3.1. Motivating example: sparsity of random blocks in the Haar basis

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 piecewise 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  $j=0, 1, \dots, J$ . Wavelet basis functions at the finest resolution level  $j=J$  are obtained by translation of the Haar mother wavelet:

$$\varphi(t) = \begin{cases} 1 & \text{if } t \in [0, 1) \\ -1 & \text{if } t \in [1, 2) \\ 0 & \text{otherwise} \end{cases}.$$

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  $\varphi_{J-1}(t) = \{1, \text{if } t \in [0, 2); -1, \text{if}$



**Fig. 1.** Random block signals (two upper) and their mixtures (two lower)

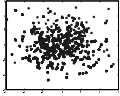
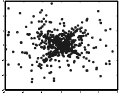
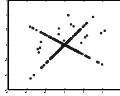
$t \in [2, 4); 0$  otherwise}. At this level, the number of non-zero coefficients still corresponds to the number of jumps, but the total number of coefficients at this level is halved, and so is the sparsity. If we further proceed to coarser resolutions, we will encounter levels where the support of 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 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.

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 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 (0.69%) – on the wavelet coefficients at the highest resolution, which have the best sparsity. Using all wavelet coefficients yields intermediate sparsity and performance.

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

**Fig. 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.

### 3.2. Multinode representation

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, such as wavelet packets and trigonometric functions libraries, might be more appropriate. The wavelet packet library consists of the triple-indexed family of functions:

$$\varphi_{j,i,q}(t) = 2^{j/2} \varphi_q(2^j t - i), \quad j, i \in \mathbf{Z}, q \in \mathbf{N}. \quad (8)$$

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  $q = 0, 1, 2, 3, \dots, 2^j - 1$  at the specified level  $j$ .

### 3.3. Adaptive selection of sparse subsets

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 (here again, the weights of the data points can be incorporated):

$$d = \sum_{k=1}^K \min_m \| u_m - x_k \|^2, \quad (9)$$

where  $K$  is the number of data points,  $u_m$  is the  $m$ -th centroid coordinates,  $x_k$  is the  $k$ -th data point coordinates, and



**Fig. 3.** Two source images (upper pair), their mixtures (middle pair) and estimated images (lower pair)

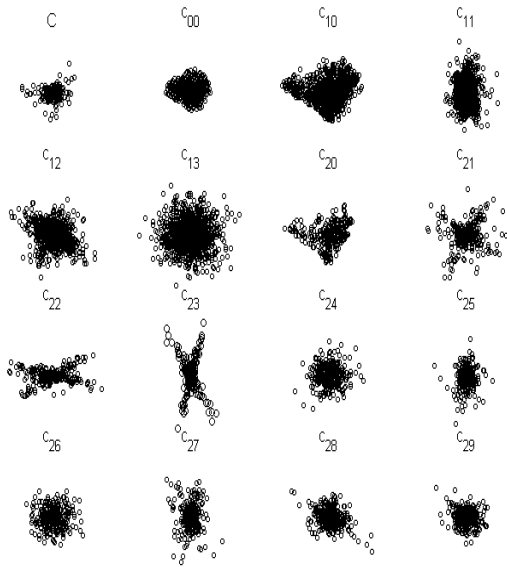
$\|\cdot\|$  is the sum-of-squares distance.

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.

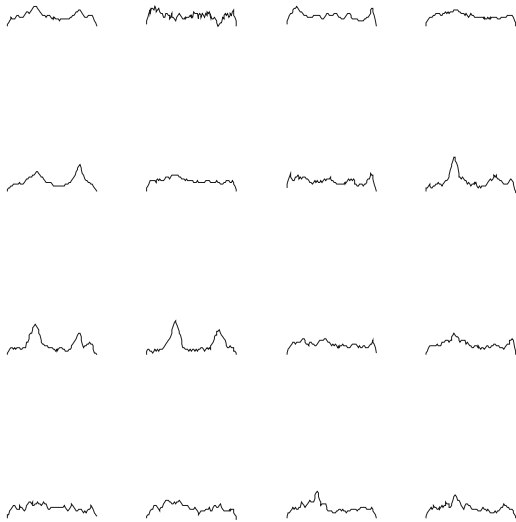
## 4. EXPERIMENTAL RESULTS

The proposed blind separation method based on the wavelet-packet representation, 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 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.



**Fig. 4.** Scatter plots of the wavelet packet (WP) coefficients of mixtures of two images; subsets are indexed on the WP tree.



**Fig. 5.** Distributions of angles (orientations) characterizing the scatter diagrams of the WP coefficients of mixtures of two images

Figure 4 shows an example of scatter plots of the wavelet packet coefficients obtained at various 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 5 shows distributions of angles (orientations) formed by points on the corresponding scatter plots of the wavelet packet coefficients at various nodes. Here, again, the sharper are the picks of a distribution, the better is the separation.

Table 1 summarizes results of experiments in which we applied our algorithm along with the FCM separation to each noise-free feature set. In these experiments we compared the quality of separation of random block and BFM signals by performing 100 Monte-Carlo simulations and calculating the normalized mean-squared errors (NMSE) for the above feature sets. (In the case of deterministic signals, we calculated a normalized squared error, NSE). In the case of image separation, we used the Discrete Cosine Transform (DCT) instead of the STFT, and the Symmlet-8 mother wavelet when using wavelet transform and wavelet packets.

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 the case of the random block signals, using the Haar wavelet function for the wavelet packet representation yields a better separation than using some smooth wavelet, e.g. Db-8. The reason is that these block signals, that are not natural signals, have a sparser representation in the case of the Haar wavelets. In contrast, as expected, natural signals such as the Flute's signals are better represented by smooth wavelets, that in turn provide a better separation. This is another advantage of using sets of features at multiple nodes along with various families of 'mother' functions: one can choose best nodes from several decomposition trees simultaneously.

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 [13].

Signals	raw	STFT	WT	WT	WP	WP
	data		db8	haar	db8	haar
Blocks	31.89	16.31	4.18	1.94	2.70	0.43
BFM sine	49.81	8.17	8.16	15.30	4.48	6.65
FM sine	50.57	5.66	10.16	24.71	4.13	5.33
Flutes	12.18	5.36	5.96	9.23	3.93	8.05

Images	raw	DCT	WT	WT	WP	WP
	data		sym8	haar	sym8	haar
	22.11	19.11	10.79	10.57	6.04	8.29

**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 with white gaussian noise	2.05	4.38	7.12	12.76	41.70
Mixtures of images with 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.

## 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.

## 6. 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] J.-F. Cardoso. "Infomax and maximum likelihood for blind separation," *IEEE Signal Processing Letters* 4 112-114, 1997.
- [5] B. A. Pearlmutter and L. C. Parra, "A context-sensitive generalization of ICA," In *ICONIP'96*, pages 151–157, 1996
- [6] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Computation*. to appear, 1998.
- [7] B. A. Olshausen and D. J. Field "Sparse coding with an overcomplete basis set: A strategy employed by v1?," *Vision Research*, 37:3311–3325, 1997.
- [8] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," In *Advances in Neural Information Processing Systems 8*. MIT Press. 1996.
- [9] S. Makeig, ICA/EEG toolbox. Computational Neurobiology Laboratory, the Salk Institute. [http://www.cnl.salk.edu/~tewon/ica\\_cnl.html](http://www.cnl.salk.edu/~tewon/ica_cnl.html), 1999.
- [10] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, 1998.
- [11] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.
- [12] *International Workshop on Independent Component Analysis and Blind Signal Separation*, (Helsinki, Finland), pp.19–20, June 2000. In press.
- [13] P. Kisilev, M. Zibulevsky, Y. Y. Zeevi, and B. A. Pearlmutter, *Multiresolution framework for sparse blind source separation*, CCIT Report no.317, June 2000