16th European Signal Processing Conference (EUSIPCO 2008), Lausanne, Switzerland, August 25-29, 2008, copyright by EURASIP

ESTIMATING THE MIXING MATRIX IN SPARSE COMPONENT ANALYSIS (SCA) USING EM ALGORITHM AND ITERATIVE BAYESIAN CLUSTERING

H. Zayyani[1], *M. Babaie-Zadeh*[1], *and C. Jutten*[2]

[1] Department of Electrical Engineering, Sharif University of Technology
Azadi Avenue, Tehran, Iran
phone: + (98) 2166165925, fax: + (98) 2166023261
email: zayyani@ee.sharif.edu, mbzadeh@yahoo.com
[2] GIPSA-lab, Dept. of Images and Signals, Institut National Polytechnique de Grenoble (INPG)
Felix viallet Avenue, Grenoble, France
phone: + (33) 476574351, fax: + (33) 476574790
email: Christian.Jutten@inpg.fr

ABSTRACT

In this paper, we focus on the mixing matrix estimation which is the first step of Sparse Component Analysis. We propose a novel algorithm based on Expectation-Maximization (EM) algorithm in the case of two-sensor set up. Then, a novel iterative Bayesian clustering is applied to yield better results in estimating the mixing matrix. Also, we compute the Maximum Likelihood (ML) estimates of the elements of the second row of the mixing matrix based on each cluster. The simulations show that the proposed method has better accuracy and less failure than the EM-Laplacian Mixture Model (EM-LMM) method.

1. INTRODUCTION

Sparse Component Analysis (SCA) [1] is a semi-blind source separation approach, in which the prior information about the sources is their sparsity. A sparse signal is a signal whose most samples are nearly zero (say they are 'inactive'), and just a few percents takes significant values (say they are 'active'). This prior information enables us to separate sources with less sensors than sources [2, 3, 4, 5, 6, 7, 8, 9]. The mathematical model of the instantaneous underdetermined Blind Source Separation (BSS) in the noisy case is:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{v}. \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times m}$ is the mixing matrix, $\mathbf{x}$ and $\mathbf{s}$ are the observation and source vectors respectively. In underdetermined case, the number of observations is less than the number of sources ($n < m$). Therefore, estimating the mixing matrix is not sufficient to recover sources, since the mixing matrix is not invertible. Therefore, underdetermined SCA consists of two steps. First, estimating the mixing matrix and then estimating the sparse sources. However, there are some methods that the mixing matrix estimation and source estimation are done simultaneously [3, 4], but in the methods that separately estimate them, the mixing matrix estimation affects the accuracy of the source estimation step. Hence, in those methods accurate mixing matrix estimation is an important preprocessing for the final source recovery stage. Therefore, in this paper, we focus on the first step.

This work has been partially supported by Iran National Science Foundation (INSF) under contract number 86/994, and by Iran Telecommunications research Center (ITRC).

Several approaches were proposed to address the mixing matrix estimation in SCA. K-means clustering [2] is a primary approach. A cluster-wise PCA method [5] and a subspace clustering method [10] are also used as later clustering methods to estimate the mixing matrix. Also, a time-frequency-transform-based clustering method were proposed in [7]. In [11], the Laplacian Mixture Model (LMM) is assumed for the distribution of $\theta = \arctan(\frac{x_2}{x_1})$ in the case of two-sensor set up. Then, an EM algorithm finds the ML estimation of the parameters of this LMM. So, this method is called EM-LMM method. Moreover, a method called soft-LOST was proposed in [12] which determine the line orientations in the scatter plot by an EM algorithm.

In this paper, we estimate the mixing matrix in the two-sensor case (similar to [11]). Our algorithm is an EM algorithm applied directly to the second observation rather than to $\theta = \arctan(\frac{x_2}{x_1})$ as in [11]. Moreover, we use an iterative Bayesian clustering approach to yield better accuracy in our estimations.

2. ESTIMATING THE MIXING MATRIX IN THE TWO-SENSOR CASE: ITERATIVE EM ALGORITHM

Consider the two-sensor case. In this case, our model in (1) can be written as:

$$x_1 = s_1 + s_2 + ... + s_m + n_1. \tag{2}$$

$$x_2 = a_1 s_1 + a_2 s_2 + ... + a_m s_m + n_2. \tag{3}$$

where x_1 and x_2 are the two sensors, n_1 and n_2 are two independent Gaussian noises with variance σ_n^2. Moreover, we assume a spiky model for the sparse sources which is a special case of Bernoulli-Gaussian (BG) distribution (this model has also been used in [8], [9]). So, the Probability Density Function (PDF) of the sources is assumed to be:

$$p(s_i) = p\delta(s_i) + (1-p)N(0, \sigma_r^2). \tag{4}$$

where p is the probability of inactivity of the sources and is near one, and σ_r^2 is the variance of active samples of the sources. We also define the source activity vector as an m-tuple vector $\mathbf{q} = (q_1, q_2, ..., q_m)^T$ where the i'th component shows the activity of the i'th source:

$$q_i \triangleq \begin{cases} 1 & \text{if } s_i \text{ is active} \\ 0 & \text{if } s_i \text{ is inactive} \end{cases}$$

In [8], we suggested a parameter estimation step, to estimate the parameters of this model, that are p, σ_r and σ_n. This parameter estimation step is firstly applied to the first mixture in (2). Then, the estimation of these parameters are available and we can use them in the rest of our algorithm in the present paper. To estimate the mixing matrix, which is equivalent to estimating the second row of the mixing matrix $(a_1, a_2, ..., a_m)$, we compute the PDF of the second sensor signal, which has a Mixture of Gaussian (MoG) density:

$$p(x_2) = \sum_{\mathbf{q}} p(\mathbf{q}) p(x_2|\mathbf{q}). \quad (5)$$

where $p(\mathbf{q}) = p^{(m-n_a)}(1-p)^{n_a}$ is the probability of $\mathbf{q}$, in which n_a is the number of active sources, and the summation is taken over 2^m possible values for the source activity vector. Since $(1-p) << 1$, the terms with $n_a > 1$ can be neglected. Therefore, the sparse approximation of the MoG distribution of the second observation is:

$$p(x_2) \approx p^m N(0, \sigma_n^2) + p^{m-1}(1-p) \sum_{i=1}^{m} N(0, a_i^2 \sigma_r^2 + \sigma_n^2). \quad (6)$$

So, the second observation has $(m+1)$ major Gaussian components. If we apply the EM algorithm of [13] to the second sensor, this EM algorithm calculates the ML estimates of the MoG components iteratively. Then, from the estimated variances, the absolute value of the second row of the mixing matrix will be obtained as follows:

$$|\widehat{a}_i| = \frac{\sqrt{\widehat{\sigma}_i^2 - \sigma_n^2}}{\sigma_r}. \quad (7)$$

where σ_r and σ_n are obtained from the parameter estimation step and $\widehat{\sigma}_i$ are obtained from the EM algorithm of [13] applied to the second observation in (6). Initial values of the parameters of the EM algorithm can be chosen randomly. We can use the final solutions of the EM algorithm for the next initialization of the parameters in the next EM algorithm. Our simulations show that the EM algorithm converges in 50 to 100 iterations.

After finding the absolute values of all a_i's, a sign ambiguity can be resolved by counting the number of observation slopes (of the observation points) around a vicinity of $\pm a_i$'s (for example in $[\pm a_i - \varepsilon \quad \pm a_i + \varepsilon]$), and then select the sign with the maximum counter.

This EM algorithm is applied only to the second observation, while the EM-LMM method is applied to the ratio of the observations which depends to both observations. So, our EM algorithm should give worse results compared to EM-LMM, and this will be verified in the simulation results. However, our simulations show the less sensitivity of our EM algorithm to initializations. It may be due to exact mixture of Gaussian model rather than heuristic mixture of Laplacian model in the EM-LMM approach.

3. ITERATIVE BAYESIAN CLUSTERING

To yield better estimation of the mixing matrix, we propose an iterative Bayesian clustering. In this method, at first the hypothesis that at least one source is active is checked with a Bayesian hypothesis testing. After detecting that at least one source is active, we assume that only one source is active and then find this unique active source (say the i'th source). Then, the i'th column is updated iteratively. The block diagram of this algorithm is shown in Fig.1.

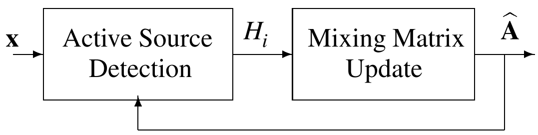

Figure 1: The block diagram of iterative Bayesian-clustering

3.1 Active source detection

In this stage, we perform a Bayesian hypothesis testing to know at which instants, at least one of sources is active. Then, assuming that only one source is active, we find which one is active. Let $H_i, i = 1, 2, ..., m$ denote the hypothesis that only the i'th source is active and H_0 is the hypothesis that all sources are inactive. We do the hypothesis testing by computing the posterior probability of $p(H_i|\mathbf{x}), i = 0, 1, ..., m$ for each hypothesis. Similar to computations in [8], the posterior probabilities for $i = 0, 1, 2, ...m$ are:

$$p(H_i|\mathbf{x}) \propto P(H_i) p(\mathbf{x}|H_i) = \frac{p^{m-1}(1-p)}{\sqrt{\det(2\pi \mathbf{Q}_{q_i})}} \exp(\frac{-1}{2} \mathbf{x}' \mathbf{Q}_{q_i}^{-1} \mathbf{x}). \quad (8)$$

where $\mathbf{Q}_{q_i} = \sigma_r^2 \mathbf{A} \mathbf{Q}_i \mathbf{A}' + \sigma_n^2 \mathbf{I}$ and $\mathbf{Q}_i = \mathrm{diag}(\mathbf{q}_i)$, and $\mathbf{q}_i, i \neq 0$ stands for the source activity vector in which only the i'th term is one (active) and the others are zero (inactive) and $\mathbf{q}_0$ is the all zero vector. Therefore, the aim of Bayesian hypothesis testing (or Bayesian clustering) is to maximize $p(H_i|\mathbf{x})$ over $H_i, i = 0, 1, ..., m$.

3.2 Updating the mixing matrix

Knowing that the i'th source is the unique active source, we can update the i'th column of the mixing matrix. We show here that $a_i^* = \frac{x_2}{x_1}$ is the Maximum Likelihood (ML) estimate of a_i in (3) under hypothesis H_i which is:

$$H_i : \begin{cases} x_1 = s_i + n_1 \\ x_2 = a_i s_i + n_2 \end{cases}$$

In fact, the ML estimate of a_i knowing all the samples in the i'th hypothesis (or cluster) is calculated in the appendix for the general case as:

$$a_i^* = \frac{\sum_k x_2(k)}{\sum_k x_1(k)}. \quad (9)$$

where $\mathbf{x} = [x_1(k), x_2(k)]^T$ is the k'th point of the i'th cluster. Hence, where only one single point is in the cluster, the above equation is equivalent to $a_i^* = \frac{x_2}{x_1}$. Unfortunately, this ML estimate is sensitive to the outliers. To show that, we assume that the ML estimate based on the first l point in the i'th cluster is $a_{i,l}^*$ and is approximately the true mixing matrix element ($a_{il}^* \simeq a_i$). Now, we consider that the $(l+1)$'th point in this cluster is an outlier where we have $\frac{x_2(l+1)}{x_1(l+1)} = a_i + e$. So, we want to prove that this outlier point changes the ML estimate significantly. To show that, we compute the ML estimation based on the first $(l+1)$ points in the cluster $(a^*_{i,l+1})$. This estimate is equal to $a^*_{i,l+1} = \frac{\sum_{k=1}^{l} x_2(k)+x_2(l+1)}{\sum_{k=1}^{l} x_1(k)+x_1(l+1)}$. After simplification based on knowing $a^*_{il} = \frac{\sum_{k=1}^{l} x_2(k)}{\sum_{k=1}^{l} x_1(k)} \simeq a_i$, we will have $a^*_{i,l+1} \simeq a_i + e\frac{x_1(l+1)}{\sum_{k=1}^{l} x_1(k)}$. The error term which is $e\frac{x_1(l+1)}{\sum_{k=1}^{l} x_1(k)}$ can be large. This proves the sensitivity of the ML estimate to the outliers. Therefore, for updating the mixing matrix, we use the iterative equation:

$$a_i^{(k+1)} = \alpha_k \frac{x_2}{x_1} + (1-\alpha_k)a_i^{(k)}. \qquad (10)$$

where we weight α_k for the current estimate based on the last single point in the cluster $(\frac{x_2}{x_1})$ and we weight $(1-\alpha_k)$ for the previous estimate based on the total previous points in the cluster. The sequence α_k should be a decreasing sequence to weight the new estimate and previous estimate. At first iterations, the α_k should be near one. But after some iterations, it should be decreased and at final iterations it can be near zero. We use $\alpha_{k+1} = \alpha_k.r$ where $r < 1$. After convergence, to track the small changes in the mixing matrix elements, we should fix the value of α_k to a small value. The value of r determines the speed of convergence. The greater the coefficient r, the smaller the rate of convergence. But, the convergence is assured. The smaller the value of r, the faster rate of convergence. But, the convergence is not guaranteed.

4. SIMULATION RESULTS

This section investigates the result of our EM algorithm in comparison with the EM-LMM algorithm in [11]. Moreover, we will see that combination of our EM algorithm with iterative Bayesian clustering results in better accuracy.

To view the results of the algorithm visually, we created two artificial instantaneous observations as in the model (2) and (3) from three artificial sparse sources. The sparse sources are created from the model (4) with parameters $p = 0.8$ and $\sigma_r = 1$. The variance of the additive Gaussian noises in (2) and (3) are selected as $\sigma_n = .01$. Also, the second row of the mixing matrix is assumed to be $[-1.5, 1, 2]$. Figure.2 shows the scatter plot of the observation points and the three clusters resulted from the iterative Bayesian method are depicted in Fig.3.

To test the algorithms quantitatively, we created two observations from four artificial sparse sources. Also, the second row of the mixing matrix is assumed to be random. The parameters of the sparse sources are $.8 \leq p \leq 0.94$ and $\sigma_r = 1$. The variance of the additive Gaussian noises are selected as $\sigma_n = .01$.

To evaluate and compare the accuracy of algorithms to estimate the mixing matrix, we define a Mean Square Error (MSE) of the mixing matrix estimation as:

$$\text{MSE}(\mathbf{A}, \hat{\mathbf{A}}) = 10\log(\frac{1}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij} - \hat{a_{ij}})^2). \qquad (11)$$

To investigate the success of the algorithms to estimate the true mixing matrix, we use the criteria $\text{MSE}(\mathbf{A}, \hat{\mathbf{A}}) < -5$. For the cases where $\text{MSE}(\mathbf{A}, \hat{\mathbf{A}}) > -5$, we call that a failure is happened and the rate of failure of each algorithm is reported.

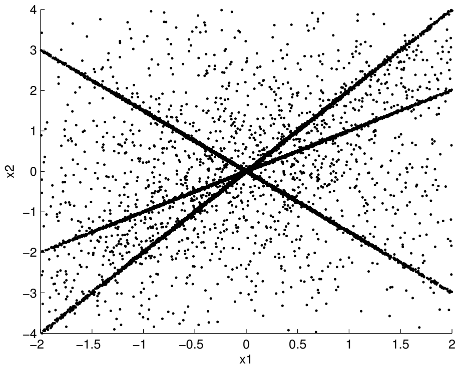

Figure 2: The scatter plot

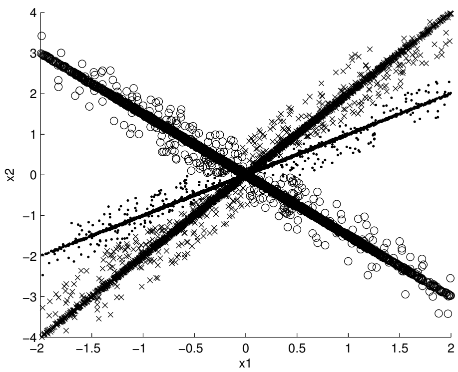

Figure 3: The three clusters from iterative Bayesian clustering

The simulations are repeated 100 times with new random sparse sources and mixing matrix, and the resulting MSE's are averaged over these 100 experiments. The result of our EM algorithm and our EM algorithm combined with iterative Bayesian clustering are compared with EM-LMM algorithm of [11].

The initialization of the EM-LMM is done with $\theta = [-3, -1, 1, 3]^T$ and $\alpha = [.25, .25, .25, .25]^T$ and $\mathbf{c} = [.5, .5, .5, .5]^T$ (refer to [11]). The initialization of our EM algorithm is done with $\sigma = [\hat{\sigma_n}, .2, .5, 2.5, 6]^T$ and $\mathbf{p} = [.2, .2, .2, .2, .2]^T$ (refer to [13]).

The results for various values of p are displayed in Fig.4. This figure shows that although the EM-LMM method has better results than our EM algorithm, the combination of our EM algorithm and iterative Bayesian clustering results in better estimation than the EM-LMM method. It is seen that the iterative Bayesian clustering improves the mixing matrix estimation.

Although the EM-LMM algorithm has better results than our EM algorithm, its failure probability is higher than our EM algorithm. In other words, its probability of not converging to the true mixing matrix is higher than our EM algorithm. The failure probability (in percent) is the number of failed experiments (MSE > -5) in all 100 experiments. The results for the failure probability is depicted in Fig.5.

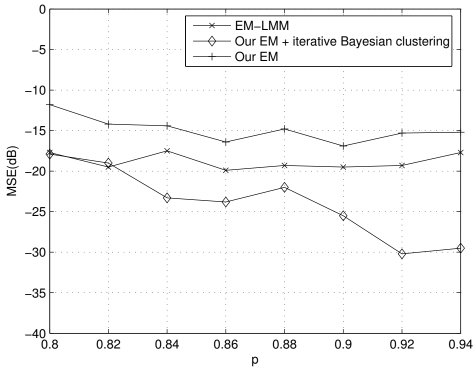

Figure 4: The MSE results for the various algorithms

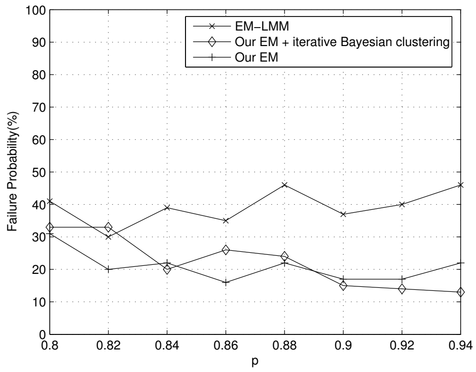

Figure 5: The failure probability for the various algorithms

This figure shows that our EM algorithm and our EM algorithm combined with iterative Bayesian clustering have less failures than the EM-LMM algorithm.

Note also that although our EM algorithm combined with iterative Bayesian clustering is relatively better than the EM-LMM algorithm (both in accuracy and failure), but in the cases of less sparse signals (for example $p \leq .82$ in Fig.4 and Fig.5), the performance of our algorithm and EM-LMM method (accuracy and failure) are relatively equivalent.

5. CONCLUSION

A new approach for estimating the mixing matrix in the two-sensor SCA problem was presented. In this approach, an EM algorithm is first applied directly to the second observation. Then, a Bayesian hypothesis testing is used to Bayesian clustering of the observation points. After the Bayesian clustering, an iterative mixing matrix estimation provides a better accuracy of estimation. This proposed algorithm shows better accuracy and less failure compared with the EM-LMM method of [11].

In the cases where we have more than two observations, there is still the mixture of Gaussian model for the observation vector in higher dimensions. So, some similar relations may exist to express the dependency of MoG parameters to the mixing matrix. But, in this case, it may be more sensitive to outliers or may be more complex to obtain mixing matrix elements from that estimated parameters.

Another approach is to apply the EM algorithm to each of the observations. But, it seems to have scaling problems and perhaps the accuracy of the estimations are low since we neglect the information of other observations. Fortunately, in the two sensor case, we consider the scaling ambiguity in our simple model to solve the problem. However, the application of the EM algorithm for mixing matrix estimation in more than two observations is a subject of investion for future works.

6. APPENDIX

We collect all the observations that lay in the i'th hypothesis in a cluster. The ML estimation of a_i should be obtained based on the observations in this cluster. This cluster or hypothesis is formulated as:

$$H_i : \begin{cases} \mathbf{x}_1 = \mathbf{s}_i + \mathbf{n}_1 \\ \mathbf{x}_2 = a_i \mathbf{s}_i + \mathbf{n}_2 \end{cases} \quad (12)$$

where $\mathbf{x}_1 = [x_1(1), \ldots, x_1(N)]^T$ and $\mathbf{x}_2 = [x_2(1), \ldots, x_2(N)]^T$ are the collection of N observations of the i'th cluster. The vector $\mathbf{s}_i = [\mathbf{s}_i(1), \ldots, \mathbf{s}_i(N)]^T$ is the collection of the i'th source for the observation points in this cluster. The likelihood based upon knowing $\mathbf{s}_i$ and a_i is computed as:

$$L_0(\mathbf{x}_1, \mathbf{x}_2 | \mathbf{s}_i, a_i) = \prod_k f_{\mathbf{n}_1 \mathbf{n}_2}(x_1(k) - s_i(k), x_2(k) - a_i s_i(k)). \quad (13)$$

Based upon the assumption of independency and Gaussianity of the noises, the likelihood is proportional to:

$$\exp\left(\frac{-1}{2\sigma_n^2} \sum_k [x_1(k) - s_i(k)]^2 + [x_2(k) - a_i s_i(k)]^2\right). \quad (14)$$

Therefore, the negative of the log likelihood can be written as (after omitting the constant terms):

$$L = -\log L_0 = \sum_k [x_1(k) - s_i(k)]^2 + [x_2(k) - a_i s_i(k)]^2. \quad (15)$$

The above equation should be minimized at ML estimate. So, setting the partial derivatives $\frac{\partial L}{\partial s_i(k)}$ and $\frac{\partial L}{\partial a_i}$ to zero, gives the ML estimate of a_i. Setting to zero the partial derivative $\frac{\partial L}{\partial s_i(k)} = -2(x_1(k) - s_i(k)) - 2a_i(x_2(k) - a_i s_i(k)) = 0$ results in the ML estimate of $s_i(k)$ as:

$$s_i^* = \frac{1}{1 + a_i^2} x_1 + \frac{a_i}{1 + a_i^2} x_2. \quad (16)$$

where the index k is omitted for simplicity. Setting to zero the partial derivative $\frac{\partial L}{\partial a_i} = \sum_k -2a_i[x_2(k) - a_i s_i(k)] = 0$ and replacing (16) for $s_i(k)$, we have the following relation for the ML estimate of a_i:

$$a_i^* = \frac{\sum_k x_2(k)}{\sum_k x_1(k)}. \quad (17)$$

REFERENCES

[1] R. Gribonval, and S. Lesage, "A survey of sparse component analysis for blind source separation: principles, prespectives, and new challanges," in *Proc. ESANN 2006*, Bruges, Belgium, April 26-28. 2006, pp. 323–330.

[2] M. Zibulevsky and B. A. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," *Neural Computation*, vol. 13, pp. 863–882, April. 2001.

[3] M. Davies and N. Mitianoudis, "Simple mixure model for sparse overcomplete ICA," *IEE Proceeding on Visual, Image and Signal Processing*, vol. 151, pp. 35–43, Jan. 2004.

[4] C. Fevotte and S. J. Godsill, "A Bayesian approach for blind separation of sparse sources," *IEE Trans On Speech and Audio Processing*, vol. 14, pp. 2174–2188, Nov. 2006.

[5] M. Babaie-zadeh, C. Jutten, and A. Mansour, "Sparse ICA via cluster-wise PCA," *Neurocomputing*, vol. 69, pp. 1458–1466, Aug. 2006.

[6] P. G. Georgiev, F. J. Theis, and A. Cichoki, "Sparse component analysis and blind source separation of underdetermined mixtures," *IEEE Trans On Neural Networks*, vol. 16, pp. 992–996, July. 2005.

[7] Y. Li, S. I. Amari, A. Cichoki, D. W. C. Ho, and S. Xie, "Underdetermined blind source separation based on sparse representation," *IEEE Trans On Signal Processing*, vol. 54, pp. 423–437, Feb. 2006.

[8] H. Zayyani, M. Babaie-zadeh, and C. Jutten, "Source estimation in noisy sparse component analysis," in *Proc. of 15'th Int. Conf. On Digital Signal Processing (DSP 2007)*, Cardiff, Wales, UK, July 1-4. 2007, pp. 219–222.

[9] L. Vielva, D. Erdogmus, and J. C. Principe, "Underdetermined blind source separation using a probabilistic source sparsity model," in *Proc. of ICA 2001*, San Diego, California, USA, December 9-13. 2001, pp. 675–679.

[10] F. M. Movahedi, G. H. Mohimani, M. Babaie-zadeh, and C. Jutten, "Estimating the mixing matrix in sparse component analysis based on partial subspace clustering," *Accepted for Neurocomputing*.

[11] N. Mitianoudis, and T. Stathaki, "Overcomplete source separation using laplacian mixture models," *IEEE Signal Processing Letters*, vol. 12, pp. 277–280, April. 2005.

[12] P. D. O'grady, and B. A. Pearlmutter, "Soft-LOST: EM on a mixture of oriented lines," in *Proc. of ICA 2004*, Granada, Spain, September 22-24. 2004, pp. 430–436.

[13] C. Tomasi, "Estimating Gaussian mixture density with EM-a tutorial," Available at `http://www.cs.duke.edu/courses/spring04/ cps196.1/handouts/EM/tomasiEM.pdf`.