# Blind Separation of Image Sources via Adaptive Dictionary Learning

Vahid Abolghasemi, *Member, IEEE,* Saideh Ferdowsi, *Student Member, IEEE,* and Saeid Sanei, *Senior Member, IEEE*

***Abstract*—Sparsity has been shown to be very useful in source separation of multichannel observations. However, in most cases the sources of interest are not sparse in their current domain and one needs to sparsify them using a known transform or dictionary. If such *a priori* about the underlying sparse domain of the sources is not available then the current algorithms will fail to successfully recover the sources. In this paper, we address this problem and attempt to give a solution via fusing the dictionary learning into the source separation. We first define a cost function based on this idea and propose an extension of the denoising method in [1] to minimize it. Due to impracticality of such direct extension we then propose a feasible approach. In the proposed hierarchical method, a local dictionary is adaptively learned for each source along with separation. This process improves the quality of the source separation even in noisy situations. In another part of the paper, we explore the possibility of adding global priors to the proposed method. The results of our experiments are promising and confirm the strength of the proposed approach.**

***Index Terms*—Blind source separation, dictionary learning, image denoising, morphological component analysis, sparsity.**

## I. INTRODUCTION

In signal and image processing, there are many instances where a set of observations is available and we wish to recover the sources generating these observations. This problem which is known as blind source separation (BSS) can be presented by the following linear mixture model:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{V}, \tag{1}$$

where $\mathbf{Y} \in \mathbb{R}^{m\times N} : \{\mathbf{y}_{i:}\}_{i=1}^{m}$ is the observation matrix, $\mathbf{X} \in \mathbb{R}^{n\times N} : \{\mathbf{x}_{i:}\}_{i=1}^{n}$ is the source matrix and $\mathbf{A}$ is the $m \times n$ mixing matrix (throughout the paper we assume that the mixing matrix is column-normalized). The additive $\mathbf{V}$ of size $m \times N$ represents the instrumental noise or the imperfection of the model. In BSS the aim is to estimate both $\mathbf{A}$ and $\mathbf{X}$ from the observations. This problem does not have a unique solution in general. However, one can find a solution for (1) by imposing some constraints into the separation process and making the sources distinguishable. Independent component analysis (ICA) is one of the well-established methods which exploits the statistical *independency* of the sources [2]. It also assumes that the sources are non-Gaussian and attempts to separate them by minimizing the mutual information.

*Nonnegativity* is another constraint which has been shown to be useful for source separation. In non-negative matrix factorization (NMF) a nonnegative matrix is decomposed into a product of two nonnegative matrices. The nonnegativity constraint allows only additive combination, so, it can produce a part-based representation of data [3].

In particular, and in image separation applications, there have been proposed several Bayesian approaches [4]–[8]. In [4], Koray *et al.* adopted a Markov random field model (MRF) to preserve the spatial dependency of neighboring pixels (e.g. sharpness of edges) in a 2-D model. They provided a fully Bayesian solution to the image separation using Gibbs sampling. Another MRF-based method has been proposed by Tonazzini *et al.* in [7]. They used maximum a posteriori (MAP) estimation method for recovering both the mixing matrix and the sources, based on alternating maximization in a simulated annealing scheme. Their experimental results have shown that this scheme is able to increase the robustness against noise. Later, Tonazzini *et al.* [8] proposed to apply mean field approximation to MRF and used expectation maximization (EM) algorithm to estimate parameters of both the mixing matrix and the sources. This method has been successfully used in document processing application. Also, in [9] the images have been modeled by hidden Markov fields with unknown parameters and a Bayesian formulation has been considered. A fast MCMC (Monte Carlo Markov Chain) algorithm has been proposed in this work for joint separation and segmentation of mixed images. Kuruoglu *et al.* [10] have studied the detection of cosmic microwave background (CMB) using image separation techniques. They exploited the spatial information in the images by modeling the correlation and interactions between pixels by Markov random fields. They proposed a MAP algorithm for estimation of the sources and the mixing matrix parameters.

Different from MRF-based approaches for CMB detection in astrophysical images, is SMICA (spectral matching ICA) method proposed by Cardoso *et al.* [5]. In SMICA, the spectral diversity of the sources is exploited. This enables SMICA to separate the sources, even if they are Gaussian and noisy, provided they have different power spectra. It is shown in [5] that this idea leads to an efficient EM algorithm, much faster than previous algorithms with non-Gaussianity assumption. An extension of this method to the wavelet domain, called wSMICA has been proposed by Moudden *et al.* [6].

Wavelet domain analysis for image separation problem has



The authors are with the Department of Computing, Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, GU2 7XH, UK (e-mail: vabolghasemi@ieee.org).

also been shown to be advantageous in some other works. In [11], Ichir *et al.* assumed three different models of the wavelet coefficients: independent Gaussian mixture model, hidden Markov tree model, and contextual hidden Markov field model. Then, they proposed an MCMC algorithm for joint blind separation of the sources and estimation of the mixing matrix and the hyper parameters. The work in [12] also uses the wavelet domain for source separation. In another work, Bronstein *et al.* [13] studied the problem of recovering a scene recorded through a semi-reflecting medium and proposed a technique called SPICA (sparse ICA). They proposed to apply the wavelet packet transform (WPT) to the mixtures prior to applying Infomax ICA. Their algorithm is suitable for the cases where all the sources can be sparsely represented in the wavelet domain.

Sparsity-based approaches for BSS problem have been received much attention recently. The term *sparse* refers to signals or images with small number of non-zeros with respect to some representation basis. In sparse component analysis (SCA) the assumption is that the sources can be sparsely represented using a known common basis or dictionary. For instance the SPICA method [13], mentioned above, considers wavelet domain for this purpose. Also the proposed methods in [14] and [15] use sparse dictionaries for separation of speech mixtures. Nevertheless, there are many cases where each source is sparse in a different domain which makes it difficult to directly apply SCA methods to their mixtures. Multichannel morphological component analysis (MMCA) [16] and generalized morphological component analysis (GMCA) [17] have been recently proposed to address this problem. The main assumption in MMCA is that each source can be sparsely represented in a specific known transform domain. In GMCA, each source is modeled as the linear combination of a number of morphological components where each component is sparse in a specific basis. MMCA and GMCA are extensions of previously proposed method of morphological component analysis (MCA) [18]–[21] to the multichannel case. In MCA the given signal/image is decomposed into different morphological components subject to sparsity of each component in a known basis (or dictionary).

MMCA performs well where prior knowledge about the sparse domain of each individual source is available; but what if such *a priori* is not available or what if the sources are not sparsely representable using the existing transforms? One may answer these questions by referring to the dictionary learning framework. In dictionary learning framework the aim is to find an overcomplete dictionary which can sparsely represent a given set of images or signals. Utilizing learned dictionaries into MCA has been shown to yield promising results [22], [23]. However, taking the advantages of learned dictionaries in MMCA is still an open issue. One possible approach to use dictionary learning for MMCA is to learn a specific dictionary for each source from a set of exemplar images. This however is rarely the case, since in most BSS problems such training samples are not available.

In this paper we adapt MMCA to those cases that the sparsifying dictionaries/transforms are not available. The proposed algorithm is designed to adaptively learn the dictionaries from the mixed images within the source separation process. This method is motivated by the idea of image denoising using a learned dictionary from the corrupted image in [1]. Other derivatives of [1] have also been reported in [22], [24], [25] for the purpose of image separation from a single mixture. In the present paper, the multichannel case with more observations than sources ($m \geq n$) is considered. We start by theoretically extending the denoising problem to BSS. Then, a practical algorithm is proposed for blind source separation without any prior knowledge about the sparse domains of the sources. The results indicate that adaptive dictionary learning, one for each source, enhances the separability of the sources.

### A. Notation

In this paper all parameters have real values even though it is not explicitly mentioned. We use small and capital bold face characters to represent vectors and matrices respectively. All vectors are represented as column-vectors by default. For instance, the $j$-th column and $j$-th row of matrix $\mathbf{X}$ are represented by column vectors $\mathbf{x}_{:j}$ and $\mathbf{x}_{j:}$, respectively. The vectorized version of matrix $\mathbf{X}$ is shown by $\underline{\mathbf{x}} = \mathrm{vec}(\mathbf{X})$. The matrix Frobenius norm is indicated by $\|\cdot\|_F$. The transpose operation is denoted by $(\cdot)^T$. The $\ell_0$-norm which counts the number of non-zeros and $\ell_1$-norm which sums over the absolute values of all elements are shown by $\|\cdot\|_0$ and $\|\cdot\|_1$, respectively.

### B. Organization of the paper

In the next section the motivation behind the proposed method is stated. We start by describing the denoising problem and its relation to image separation. Then, the extension of single mixture separation to the multichannel observation scenario is argued. In section III a practical approach for this purpose is proposed. Section IV is devoted to discussing the possibility of adding global sparsity priors to the proposed method. Some practical issues are discussed in section V. The numerical results are given in section VI. Finally, the paper is concluded in section VII.

## II. From Image Denoising to Source Separation

### A. Image denoising

Consider a noisy image corrupted by additive noise. Elad *et al.* [1] showed that if the knowledge about the noise power is available, it is possible to denoise it by learning a local dictionary from the noisy image itself. In order to deal with large images they used small (overlapped) patches to learn such dictionary. The obtained dictionary is considered *local* since it describes the features extracted from small patches. Let us represent the noisy $\sqrt{N} \times \sqrt{N}$ image $\mathcal{Y}$ by vector $\mathbf{y}$ of length $N$. The unknown denoised image $\mathcal{X}$ is also vectorized and represented as $\mathbf{x}$. The $i$-th $\sqrt{r} \times \sqrt{r}$ patch from $\mathcal{X}$ is shown by vector $\mathfrak{R}_i\mathbf{x}$ of $r \ll N$ pixels. For notational simplicity the $i$-th patch is expressed as explicit multiplication of operator $\mathfrak{R}_i$ (a binary $r \times N$ matrix) by $\mathbf{x}$.[1] The overall denoising problem is expressed as

[1]Practically, we apply a nonlinear operation to extract the patches from image $\mathcal{X}$, by sliding a mask of appropriate size over the entire image, similar to [1] and [22].

$$\min_{\mathbf{D},\{\mathbf{s}_i\},\mathbf{x}} \lambda \|\mathbf{y}-\mathbf{x}\|_2^2 + \sum_i \mu_i \|\mathbf{s}_i\|_0 + \sum_i \|\mathbf{D}\mathbf{s}_i - \mathfrak{R}_i\mathbf{x}\|_2^2, \quad (2)$$

where scalars $\lambda$ and $\mu$ control the noise power and sparsity degree, respectively. Also, $\mathbf{D} \in \mathbb{R}^{r\times k}$ is the sparsifying dictionary which contains normalized columns (also called atoms) and $\{\mathbf{s}_i\}$ are sparse coefficients of length $k$.

In the proposed algorithm by Elad *et al.* [1], $\mathbf{x}$ and $\mathbf{D}$ are respectively initialized with $\mathbf{y}$ and overcomplete ($r < k$) discrete cosine transform (DCT) dictionary. The minimization of (2) starts with extracting and re-arranging all the patches of $\mathbf{x}$. The patches are then processed by K-SVD [26] which updates $\mathbf{D}$ and estimates the sparse coefficients $\{\mathbf{s}_i\}$. Afterward, $\mathbf{D}$ and $\{\mathbf{s}_i\}$ are assumed fixed and $\mathbf{x}$ is estimated by computing

$$\hat{\mathbf{x}} = \left(\lambda\mathbf{I} + \sum_i \mathfrak{R}_i^T\mathfrak{R}_i\right)^{-1}\left(\lambda\mathbf{y} + \sum_i \mathfrak{R}_i^T\mathbf{D}\mathbf{s}_i\right), \quad (3)$$

where $\mathbf{I}$ is the identity matrix and $\hat{\mathbf{x}}$ is the refined version of $\mathbf{x}$. Again, $\mathbf{D}$ and $\{\mathbf{s}_i\}$ are updated by K-SVD but this time using the patches from $\hat{\mathbf{x}}$ which are less noisy. Such conjoined denoising and dictionary adaptation is repeated to minimize (2). In practice, (3) is obtained computationally easier since $\sum_i \mathfrak{R}_i^T\mathfrak{R}_i$ is diagonal and the above expression can be calculated in a pixel-wise fashion.

It is seen that (3) is a kind of averaging using both noisy and denoised patches, which if repeated along with updating of other parameters, will denoise the entire image [1]. However, in the sequel we will try to find out if this strategy is extendable for the cases where the noise is added to the mixtures of more than one image.

### B. Image separation

Image separation is a more complicated case of image denoising where more than one image are to be recovered from a single observation. Consider a single linear mixture of two textures with additive noise: $\mathcal{Y} = \mathcal{X}_1 + \mathcal{X}_2 + \mathcal{V}$ (or $\mathbf{y} = \mathbf{x}_1 + \mathbf{x}_2 + \mathbf{v}$). The authors of [25] attempt to recover $\mathbf{x}_1$ and $\mathbf{x}_2$ using the prior knowledge about two sparsifying dictionaries $\mathbf{D}_1$ and $\mathbf{D}_2$. They use a minimum mean squared error (MMSE) estimator for this purpose. In contrast, the recent work in [24] does not assume any prior knowledge about the dictionaries. It rather attempts to learn a single dictionary from the mixture and then applies a decision criterion to the dictionary atoms to separate the images. In another recent work, Peyre *et al.* [22] presented an adaptive MCA scheme by learning the morphologies of image layers. They proposed to use both adaptive *local* dictionaries and fixed *global* transforms (e.g. wavelet, curvelet) for image separation from a single mixture. Their simulation results show the effects of adaptive dictionaries on separation of complex texture patterns from natural images.

All the related studies have demonstrated the advantages that adaptive dictionary learning can have on the separation task. However, there is still one missing piece and that is considering such adaptivity for the multichannel mixtures. Performing the idea of learning local dictionaries within the source separation of multichannel observations obviously has many benefits in different applications. Next, we extend the denoising method of [1] for this purpose.

### C. Multichannel source separation

In this section the aim is to extend the denoising problem (2) to the multichannel source separation. Consider the BSS model introduced in section I, and assume further that the sources of interest are 2-D grayscale images. The BSS model for 2-D sources can be represented by vectorizing all images $\{\mathcal{X}_1, \mathcal{X}_2, \dots \mathcal{X}_n\}$ to $\{\mathbf{x}_1, \mathbf{x}_2, \dots \mathbf{x}_n\}$ and then stacking them to form $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots \mathbf{x}_n]^T$. The BSS model (1) cannot be directly incorporated into (2) as it requires both $\mathbf{X}$ and $\mathbf{Y}$ to be single vectors. Hence, we use the vectorized versions of these matrices in model (1) which can be obtained using the properties of the Kronecker product[2]:

$$\underline{\mathbf{y}} = (\mathbf{I}\otimes\mathbf{A})\,\underline{\mathbf{x}} + \underline{\mathbf{v}}. \quad (4)$$

In the above expression $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$ are column vectors of length $nN$ and $mN$, respectively. Also, $\underline{\mathbf{v}}$ is a column vector of length $mN$. $(\mathbf{I}\otimes\mathbf{A})$ is a block diagonal matrix of size $mN\times nN$, and $\otimes$ is the Kronecker product symbol. We consider the noiseless setting and modify (2) to

$$\min_{\mathbf{A},\mathbf{D},\{\mathbf{s}_i\},\underline{\mathbf{x}}} \lambda \left\|\underline{\mathbf{y}} - (\mathbf{I}\otimes\mathbf{A})\,\underline{\mathbf{x}}\right\|_2^2 + \sum_i \mu_i \|\mathbf{s}_i\|_0 + \sum_i \|\mathbf{D}\mathbf{s}_i - \mathfrak{R}_i\underline{\mathbf{x}}\|_2^2. \quad (5)$$

It is clearly seen that the above expression is similar to (2) but with an extra mixing matrix $\mathbf{A}$ to be estimated. In addition, the vectors $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$ are much lengthier than $\mathbf{x}$ and $\mathbf{y}$ as they represent vectorized versions of multiple images and mixtures.

The problem (5) can be minimized in an alternating scheme by keeping all but one unknown fixed at a time. The estimation of $\mathbf{D}$ and $\{\mathbf{s}_i\}$ can be achieved using K-SVD similar to [1]. In K-SVD the dictionary is updated column-wise using singular value decomposition (SVD); the sparse coefficients are estimated using one of the common sparse coding techniques. However, estimation of $\underline{\mathbf{x}}$ is slightly different from that of $\mathbf{x}$ in (3) and needs more attention. It has a closed form solution which is obtained by taking the gradient of (5) and setting it to zero:

$$0 = \lambda\,(\mathbf{I}\otimes\mathbf{A})^T\left((\mathbf{I}\otimes\mathbf{A})\,\underline{\mathbf{x}} - \underline{\mathbf{y}}\right) + \sum_i \mathfrak{R}_i^T\left(\mathfrak{R}_i\underline{\mathbf{x}} - \mathbf{D}\mathbf{s}_i\right), \quad (6)$$

leading to:

$$\underline{\hat{\mathbf{x}}} = \left(\lambda\,(\mathbf{I}\otimes\mathbf{A})^T(\mathbf{I}\otimes\mathbf{A}) + \sum_i \mathfrak{R}_i^T\mathfrak{R}_i\right)^{-1} \times \left(\lambda\,(\mathbf{I}\otimes\mathbf{A})^T\underline{\mathbf{y}} + \sum_i \mathfrak{R}_i^T\mathbf{D}\mathbf{s}_i\right). \quad (7)$$

[2]The matrix multiplication can be expressed as a linear transformation on matrices. In particular: $\mathrm{vec}(\mathbf{ABC}) = (\mathbf{C}^T\otimes\mathbf{A})\mathrm{vec}(\mathbf{B})$ for three matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. In our case, $\mathrm{vec}(\mathbf{AX}) = (\mathbf{I}\otimes\mathbf{A})\mathrm{vec}(\mathbf{X})$.

In order to estimate $\mathbf{A}$ we consider all unknowns, except $\mathbf{A}$, fixed and simplify (5) by converting the first quadratic term into ordinary matrix product and obtain:

$$\min_{\mathbf{A}} \lambda \left\| \mathbf{Y} - \mathbf{A}\mathbf{X} \right\|_F^2 . \tag{8}$$

The above minimization problem is easily solved using pseudo-inverse of $\mathbf{X}$ as:

$$\hat{\mathbf{A}} = \mathbf{Y}\mathbf{X}^T \left( \mathbf{X}\mathbf{X}^T \right)^{-1} . \tag{9}$$

The above steps (i.e. estimation of $\mathbf{D}$, $\{\mathbf{s}_i\}$, $\underline{\mathbf{x}}$ and $\mathbf{A}$) should be alternately repeated to minimize (5). However, the long expression (7) is not practically computable, especially if the number of sources and observations is large. That is because of dealing with huge $mN \times nN$ matrix $(\mathbf{I} \otimes \mathbf{A})$. In addition, in contrast to the aforementioned denoising problem, the matrix to be inverted in (7) is not diagonal and, so, the estimation of $\underline{\mathbf{x}}$ cannot be calculated pixel-by-pixel, which makes the situation more difficult to handle. In the next section a practical approach is proposed to solve this problem.

## III. Algorithm

In order to find a practical solution for (5) we use a hierarchical scheme such as the one in MMCA [16]. To do this, the BSS model (1) is broken into $n$ rank-1 multiplications and the following minimization problem is defined:

$$\min_{\{\mathbf{a}_{:j}, \mathbf{D}_j, \{\mathbf{s}_i\}, \mathbf{x}_{j:}\}} \lambda_j \left\| \mathbf{E}_j - \mathbf{a}_{:j} \mathbf{x}_{j:}^T \right\|_F^2 + \sum_i \mu_i \left\| \mathbf{s}_i \right\|_0 + \sum_i \left\| \mathbf{D}_j \mathbf{s}_i - \mathfrak{R}_i \mathbf{x}_{j:} \right\|_2^2, \tag{10}$$

where $\mathbf{D}_j$ denotes the dictionary corresponding to $j$-th source, i.e. $\mathbf{x}_{j:}$, and $\mathbf{E}_j$ is the $j$-th residual expressed as:

$$\mathbf{E}_j = \mathbf{Y} - \sum_{l=1,\, l \neq j}^{n} \mathbf{a}_{:l} \mathbf{x}_{l:}^T . \tag{11}$$

It is important to note that with this new formulation we have to learn $n$ dictionaries, one for each source, different from (5). The advantage of this scheme is learning adaptive source-specific dictionaries, which improves source diversity, as also seen in MMCA using known transform domains [21]. In addition, the cost function in (10) does not require dealing with cumbersome matrices and vectors (as in (5)) and allows to calculate the image sources in a pixel-wise fashion.

The solution for (10) is achieved using an alternating scheme. The minimization process for $j$-th level of hierarchy can be expressed as follows. The image patches are extracted from $\mathbf{x}_{j:}$ and then processed by K-SVD for learning $\mathbf{D}_j$ and sparse coefficients $\{\mathbf{s}_i\}$, while other parameters are kept fixed. Then, the gradient of (10) with respect to $\mathbf{x}_{j:}$ is calculated and set to zero:

$$0 = \lambda_j \left( \mathbf{x}_{j:} \mathbf{a}_{:j}^T - \mathbf{E}_j^T \right) \mathbf{a}_{:j} + \sum_i \mathfrak{R}_i^T \left( \mathfrak{R}_i \mathbf{x}_{j:} - \mathbf{D}_j \mathbf{s}_i \right) = \lambda_j \mathbf{x}_{j:} + \sum_i \mathfrak{R}_i^T \mathfrak{R}_i \mathbf{x}_{j:} - \lambda_j \mathbf{E}_j^T \mathbf{a}_{:j} - \sum_i \mathfrak{R}_i^T \mathbf{D}_j \mathbf{s}_i . \tag{12}$$

Finally, after some manipulations and simplifications on (12), the estimation for $j$-th source is obtained as:

$$\hat{\mathbf{x}}_{j:} = \left( \lambda_j \mathbf{I} + \sum_i \mathfrak{R}_i^T \mathfrak{R}_i \right)^{-1} \left( \lambda_j \mathbf{E}_j^T \mathbf{a}_{:j} + \sum_i \mathfrak{R}_i^T \mathbf{D}_j \mathbf{s}_i \right) . \tag{13}$$

It is interesting to notice that the inverting term in the above expression is the same as that in (3) for the denoising problem. Thus, as mentioned before, this calculation can be obtained pixel-wise. Next, in order to update $\mathbf{a}_{:j}$ a simple least square linear regression such as the one in [16] will give the solution:

$$\hat{\mathbf{a}}_{:j} = \mathbf{E}_j \mathbf{x}_{j:} . \tag{14}$$

However, normalization of $\hat{\mathbf{a}}_{:j}$ is necessary after each update to preserve the column norm of mixing matrix. The above steps for updating all variables are executed for all $j$ from $1$ to $n$. Moreover, the entire procedure should be repeated to minimize (10). A pseudo-code of the proposed algorithm is given in Algorithm 1.

As already implied, the motivation behind the proposed algorithm is to learn source-specific dictionaries offering sparse representations. Such dictionary learning, when embedded into

**Algorithm 1:** The proposed algorithm.

**Input**: Observation matrix $\mathbf{Y}$, patch size $r$, Number of dictionary atoms $k$, number of sources $n$, noise standard deviation $\sigma_v$, noise gain $C$, regularization parameter $\lambda$ and total number of iterations $M$.
**Output**: Dictionaries $\mathbf{D}_j$, sparse coefficients $\{\mathbf{s}_i\}$, sources matrix $\mathbf{X}$, and mixing matrix $\mathbf{A}$.

```
begin
  Initialization:
    • Set {D_j}_{j=1}^n to overcomplete DCT;
    • Set A to a random column-normalized matrix;
    • X = A^T Y;
    • Choose σ to be multiple times of σ_v;
    • Δσ = (1/M)(σ − σ_v);
  repeat
    λ = 30/σ;
    for j=1 to n do
      Extract all the patches from x_{j:};
      Solve the sparse recovery problem:
      ∀i min_{s_i} ||s_i||_0  s.t. ||D_j s_i − ℜ_i x_{j:}||_2^2 ≤ (Cσ)^2;
      Update D_j using K-SVD;
      Calculate the residual: E_j = Y − Σ_{l≠j} a_{:l} x_{l:}^T;
      Compute x_{j:} using (13);
      a_{:j} = E_j x_{j:};
      a_{:j} ← a_{:j} / ||a_{:j}||_2;
    end
    σ ← σ − Δσ;
  until stopping criterion is met;
end
```

the source separation process, improves both separation and dictionary learning quality. In other words, in the first few iterations of Algorithm 1 each source includes portions of other sources. However, the dictionaries gradually learn the dominant components and reject the weak portions caused by other sources. Using these dictionaries, the estimated sources—which are used for dictionary learning in the next iteration—will have less amount of irrelevant components. This adaptive process is repeated until most of the irrelevant portions are rejected and dictionaries become source-specific. The entire above procedure is carried out to minimize the cost function in (10). It is also noteworthy to mention that (10) is not jointly convex in all parameters. Therefore, the alternating estimation of these parameters, using the proposed method, does not necessarily lead to an optimal solution. This is the case for other previous alternating minimization problems too [16], [24]. However, the obtained parameters are local minima of (10) and, hence, are considered as approximate solutions.

## IV. Global Dictionaries vs. Local Dictionaries

The proposed method takes the advantages of fully *local* dictionaries which are learned within the separation task. We call these dictionaries *local*, since they capture the structure of small image patches to generate dictionary atoms. In contrast, the *global* dictionaries are generally applied to the entire image or signal and capture the global features. Incorporating the global fixed dictionaries into the proposed method can be advantageous where a prior knowledge about the structure of sources is available. Such combined *local* and *global* dictionaries has been used in [22] for single mixture separation. Here, we consider the multichannel case and extend our proposed method in section III for this purpose.

Consider a known global unitary $N \times N$ basis $\mathbf{\Phi}_j$ for each source. The minimization problem (10) can be modified as follows to capture both global and local structures of the sources:

$$\min_{\{\mathbf{a}_{:j},\mathbf{D}_j,\{\mathbf{s}_i\},\mathbf{x}_{j:}\}} \lambda_j \left\|\mathbf{E}_j - \mathbf{a}_{:j}\mathbf{x}_{j:}^T\right\|_F^2 + \beta_j \left\|\mathbf{x}_{j:}^T\mathbf{\Phi}_j\right\|_1 + \sum_i \mu_i \|\mathbf{s}_i\|_0 + \sum_i \|\mathbf{D}_j\mathbf{s}_i - \mathfrak{R}_i\mathbf{x}_{j:}\|_2^2. \quad (15)$$

Note that the term $\left\|\mathbf{x}_{j:}^T\mathbf{\Phi}_j\right\|_1$ is exactly similar to what is used in the original MMCA [16]. All variables in the above expression can be similarly estimated using Algorithm 1, except the actual sources $\{\mathbf{x}_{j:}\}$. In order to find $\mathbf{x}_{j:}$, the gradient of (15) with respect to $\mathbf{x}_{j:}$ is set to zero, leading to:

$$\left(\lambda_j\mathbf{I} + \sum_i \mathfrak{R}_i^T\mathfrak{R}_i\right)\mathbf{x}_{j:} = \underbrace{\left(\lambda_j\mathbf{E}_j^T\mathbf{a}_{:j} + \sum_i \mathfrak{R}_i^T\mathbf{D}_j\mathbf{s}_i\right)}_{\tilde{\mathbf{x}}_{j:}} - \frac{\beta_j}{2}\mathbf{\Phi}_j\mathrm{sgn}(\mathbf{\Phi}_j^T\mathbf{x}_{j:}), \quad (16)$$

where $\mathrm{sgn}(.)$ is a componentwise signum function. The above expression amounts to soft-thresholding [16] due to the signum function, and hence, the estimation of $\mathbf{x}_{j:}$ can be obtained by the following steps:

- Soft-thresholding of $\boldsymbol{\alpha}_j \triangleq \mathbf{\Phi}_j^T\tilde{\mathbf{x}}_{j:}$ with the threshold $\beta_j$ and attaining $\hat{\boldsymbol{\alpha}}_j$,
- Reconstructing $\mathbf{x}_{j:}$ by:

$$\hat{\mathbf{x}}_{j:} = \left(\lambda_j\mathbf{I} + \sum_i \mathfrak{R}_i^T\mathfrak{R}_i\right)^{-1}\mathbf{\Phi}_j\hat{\boldsymbol{\alpha}}_j. \quad (17)$$

Note that since $\mathbf{\Phi}_j$ is a unitary and known matrix, it is not explicitly stored but implicitly applied as a forward or inverse transform, where applicable. Similar to previous section, the above expression can be executed pixel-wise and is not computationally expensive.

## V. Practical Issues

Implementation of the proposed methods needs some careful considerations which are addressed here.

### A. Noise

Similar to the denoising methods in [1], the proposed method also requires the knowledge about noise power. This should be utilized in the sparse coding step of the K-SVD algorithm for solving the following problem:

$$\forall i \quad \min_{\mathbf{s}_i} \|\mathbf{s}_i\|_0 \quad s.t. \quad \|\mathbf{D}_j\mathbf{s}_i - \mathfrak{R}_i\mathbf{x}_{j:}\|_2^2 \leq (C\sigma)^2, \quad (18)$$

where $C$ is a constant and $\sigma$ is the noise standard deviation. The above well-known sparse recovery problem can be solved using orthogonal matching pursuit (OMP) [27]. The Focal Under-determined System Solver (FOCUSS) [28] method is also used to solve (18) by replacing $\ell_0$-norm with $\ell_1$-norm.

Furthermore, incorporating the noise power in solving (18) has an important advantage in the dictionary update stage. It ensures that the dictionary does not *learn* the existing noise in the patches. Consequently, the estimated image using this dictionary would become cleaner which will later refine the dictionary atoms in the next iteration. This progressive denoising loop is repeated until a clean image is achieved.

In the proposed method, however, the *noise* means the portions of other sources which are mixed with $\mathbf{x}_{j:}$. Hence, we initially consider a high value for $(C\sigma)^2$, since the portions of other sources might be high even though the noise is zero, and then gradually reduce it to zero as the iterations evolve. Nevertheless, if the observations themselves are noisy, then, we should start from a higher bound and decrease it toward the actual noise power as the iterations evolve.

### B. Dictionary and patch sizes

Unfortunately, not much can be theoretically said about choosing the optimum dictionary size (or redundancy factor: $k/r$). Highly overcomplete dictionaries allow higher sparsity of the coefficients, though, being computationally expensive. However, it may not be necessary to choose a very high redundancy and it depends on the data type and number of training signals [24]. There have been some reported works allowing a variable dictionary size to find the optimum redundancy factor such as in [29], [30]. Such analysis is out of the scope of this paper and we choose a fixed redundancy factor in our simulations.

Patch size is normally chosen depending on the entire image size and also the knowledge (if available) about the patterns of the actual sources. However, adopting very large patches should be avoided as they lead to large dictionaries and also provide few training signals for the dictionary learning stage. We have chosen $8 \times 8$ patches for all the experiments which seems to become a standard in the literature.

Furthermore, we normally use overlapping patches as shown to give better results in our simulations and also in the corresponding literature [1], [25]. However, we have empirically observed that the percentage of overlapping can be adjusted based on the noise level. In noiseless settings, less percentage of overlap (e.g. $50\%$) would be sufficient, while in noisy situations full overlapped patches results in the desired performance. Full overlapped patches are obtained by one-pixel shifting of the operator $\mathfrak{R}$, in both vertical and horizontal directions, over the entire image.

### C. Complexity analysis

The proposed algorithm is more computationally expensive than standard MMCA. It includes K-SVD which imposes two extra steps of sparse coding and dictionary update. However, the complexity of computing $\mathbf{x}$ is almost the same as in MMCA since (13) can be computed pixel-wise. The detailed analysis of the complexity of the proposed method per iteration is as follows. Consider lines 12 and 13 in Algorithm 1, and assume further that we use Approximate K-SVD [31]. Based on [31] these two lines cost $\mathcal{O}(N(t^2k + 2rk))$, where $t = \|\mathbf{s}_i\|_0$, for each $i$. The complexity of computing the residual (line 14) is $\mathcal{O}((n-1)mN)$. Lines 15 and 16 respectively cost $\mathcal{O}(m^2N + 2N + Rrk)$ and $\mathcal{O}(mN)$, where $R$ denotes the number of extracted patches. Finally the normalization in line 17 costs $\mathcal{O}(2m)$. Since all these calculations are executed for each source (i.e. $j = 1 \ldots n$), then, the total computation cost per each iteration of the algorithm would be: $\mathcal{O}(nN(t^2k+2rk)+n^2mN+nm^2N+2nN+nRrk+2nm)$. It is seen that due to learning one dictionary for each individual source, the proposed algorithm would be computationally demanding for large scale problems.

A possible approach to speed up the algorithm is to learn one dictionary for all the sources and then choose relevant atoms for estimating each source. This strategy however has not been completed yet and is under development. In addition, considering faster dictionary learning algorithms rather than K-SVD can alleviate the problem.

## VI. Results

In the first experiment we illustrate the results of applying Algorithm 1 to the mixtures of four sources. A severe case of image sources with different morphologies was chosen to examine the performance of the proposed method. A $6 \times 4$ full-rank random column-normalized $\mathbf{A}$ was used as the mixing matrix. Five hundred iterations was selected as the stopping

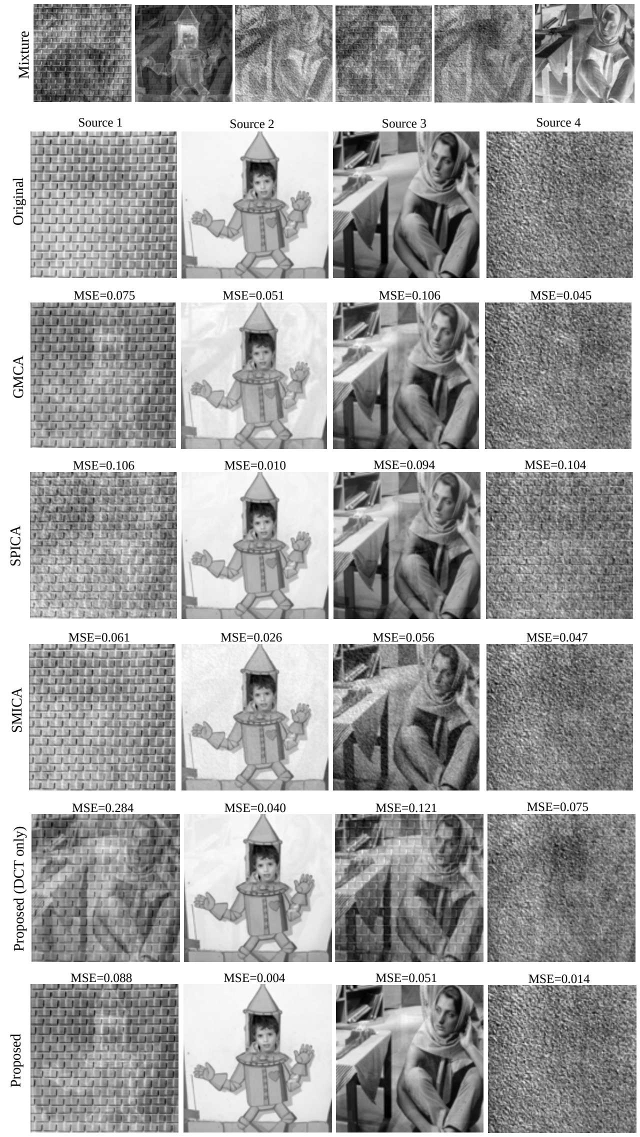


Fig. 1. The results of applying different methods to six noiseless mixtures.

criterion. The mixtures were noiseless. However, we selected $\lambda = 30/\sigma$ (see below for the reasoning of this choice) and used a decreasing $\sigma$ starting from $10$ and reaching to $0.01$ at the end of iterations. The patches had $50\%$ overlap. Other parameters were chosen as follows: $r = 64$, $k = 256$, $N = 128 \times 128$, and $C = 1$. In addition, in order to show the advantages of learning adaptive local dictionaries over the fixed local dictionaries we applied Algorithm 1 to the same mixtures, while ignoring the dictionary learning part. This way, the source separation is performed using local fixed DCT dictionaries for all sources. We also applied the following methods for comparison purposes: GMCA[3] [21] based on wavelet transform, SPICA[4] [13] based on wavelet packet transform, and SMICA [5] based on Fourier transform. Figure 1 illustrates the results of applying all these methods together with the corresponding mean squared errors (MSEs).

[3]Available at: http://md.cosmostat.org/Home.html
[4]Available at: http://visl.technion.ac.il/bron/spica/

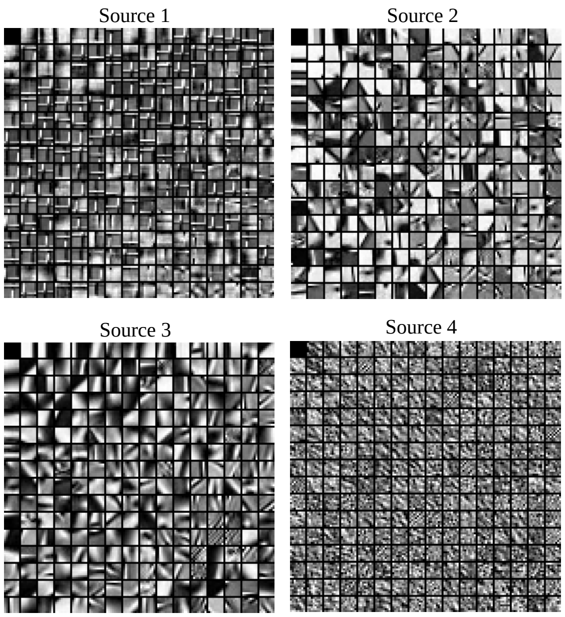


Fig. 2. The obtained dictionaries as a result of applying the proposed method to the mixtures shown in Figure 1.

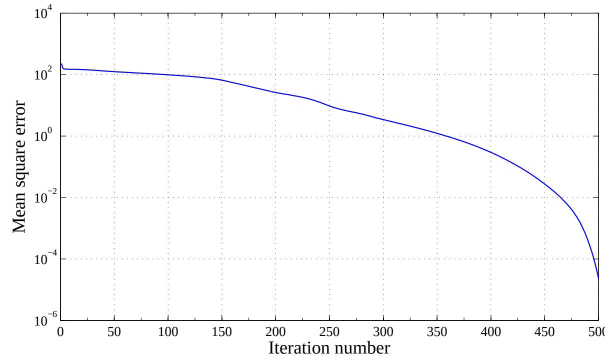


Fig. 3. Mean square error computed as $\frac{1}{mN}\|\mathbf{Y}-\mathbf{AX}\|_F^2$. Note that the elements of image sources have amplitude in the range $[0\ \ 255]$.

MSE is calculated as $\mathrm{MSE} = \frac{1}{N}\|\mathcal{X}-\hat{\mathcal{X}}\|_F^2$, where $\mathcal{X}$ and $\hat{\mathcal{X}}$ are the original and the recovered images, respectively. Figure 1, the bottom row, shows that the proposed method could successfully recover the image sources via adaptive learning of the sparsifying dictionaries. The achieved MSEs, given in Figure 1, are lowest for the proposed method for all image sources except for the *brick-like* texture. In addition, the learned dictionary atoms, shown in Figure 2, indicate good adaptation to the corresponding sources. Other methods do not perform as well as the proposed method. For instance, as seen from Figure 1, SPICA has problem in separation of *noise-like* texture and *Barbara*. GMCA has separated all image sources with some interferences from other sources. SMICA has perfectly recovered the *brick-like* texture with lowest MSE among all other methods. However, it has some difficulties in recovering *Barbara* and *cartoon boy* images.

As another measure on the performance of the proposed method, the reconstruction error, as a function of the number of iterations is shown in Figure 3. It is seen from this figure that a monotonic decrease in the value of separation error is achieved and the asymptotic error is almost zero.

In the next experiment, the performance of the proposed method in noisy situations is evaluated. For this purpose, we generated four mixtures from two image sources: *Lena* and *boat*. Then, Gaussian noise with standard deviation of $15$ was added to the mixtures so that the PSNRs (Peak Signal to Noise Ratios) of the mixtures equaled $20$ dB (Note that the image size is $128 \times 128$). The noisy mixtures are shown in Figure 4 (a). We applied the proposed algorithm to these mixtures, starting with $\sigma = 25$ and evolving while $\sigma$ was gradually decreasing to $\sigma = 15$. We used fully overlapped patches and $200$ iterations, with rest of parameters similar to those in the first experiment. One of the considerable advantages of the proposed method is the ability to denoise the sources during the separation. This is because the core idea of the proposed method comes from a denoising task. In contrast, in most conventional BSS methods the denoising should be carried out either before or after the separation, which is not ideal and may lead the algorithm to fail. Among the existing related methods, GMCA [21] uses a thresholding scheme to denoise the image sources during the separation. Hence, we applied this algorithm to the same mixtures to compare its performance with that of the proposed method. We applied SPICA and SMICA to the same mixtures too. The results of this experiment are demonstrated in Figures 4. It is seen from Figure 4 (f) that both separation and denoising tasks have been successful in the proposed method. The corresponding learned dictionaries using the proposed method, given in Figure 5 (a), confirm these observations. In addition, the proposed method is superior to GMCA, SPICA and SMICA, and denoises the sources with higher PSNRs as seen by comparing Figures 4 (c), (d), (e) and (f). Although GMCA, SPICA, and SMICA have separated the image sources, they are not successful in denoising them.

In addition, the sparsity rate of $\mathbf{s}_i$'s, obtained by averaging over the support of all $\{\mathbf{s}_i\}_{i=1}^R$ ($R$ is the number of patches), for both sources is given in Figure 5 (b). Note that a $4\times 2$ mixing matrix $\mathbf{A}$ was used in this experiment. This graph shows the percentage of used atoms by sparse coefficients, sorted in descending order, for both dictionaries corresponding to *Lena* and *boat*. As found from Figure 5 (b), the most frequently used atom shows a small percentage of appearance ($\approx 17\%$), and other dictionary atoms have been used less frequently. This means that the learned dictionaries are so efficient that few linear combinations of their atoms are sufficient to generate the image sources of interest.

In another experiment we evaluated and compared the influences of adding global sparsity prior to the proposed algorithm. The experiment settings are as follows. Two images, *texture* and *Lena*, were mixed together using a $4 \times 2$ random $\mathbf{A}$. Algorithm 1 which only takes advantage of locally learned dictionaries was applied to the mixtures. The proposed algorithm in section IV (local and global) was also applied to the same mixtures. We considered discrete wavelet transform (DWT) and DCT as global sparsifying bases for

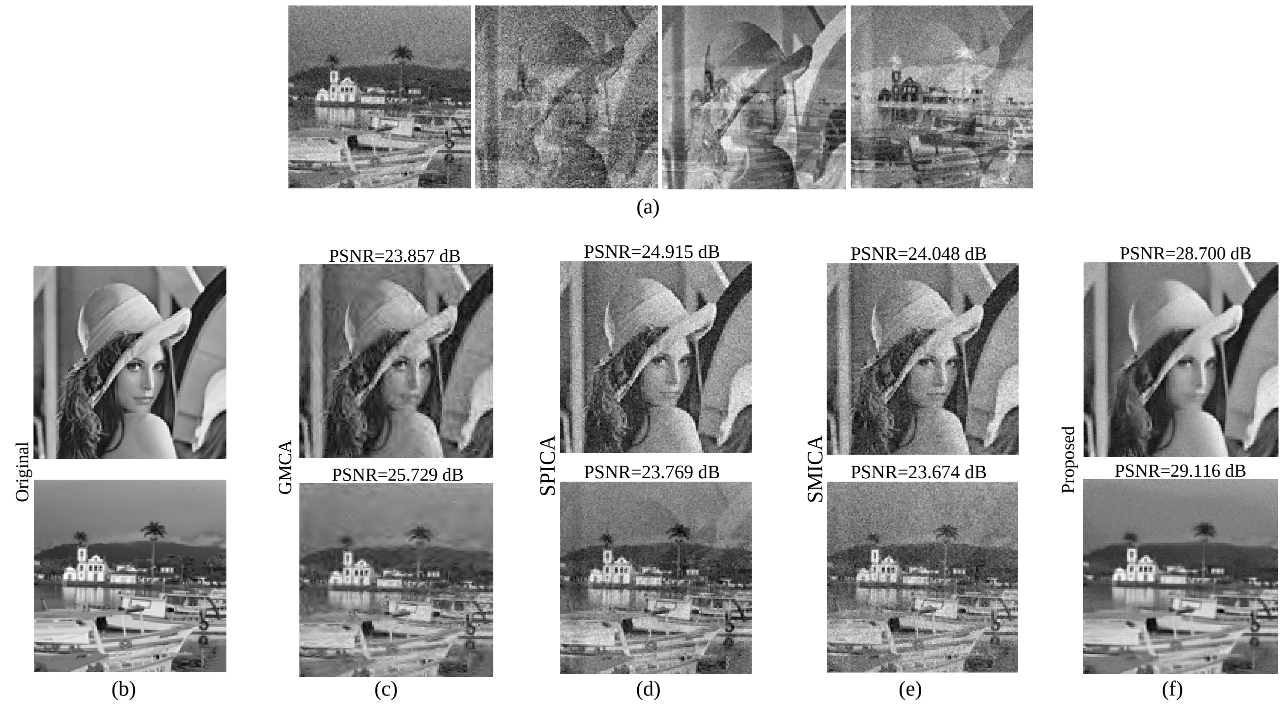


Fig. 4. (a) Noisy mixtures, (b) original sources. The separated sources using (c) GMCA, (d) SPICA, (e) SMICA, and (f) the proposed method.

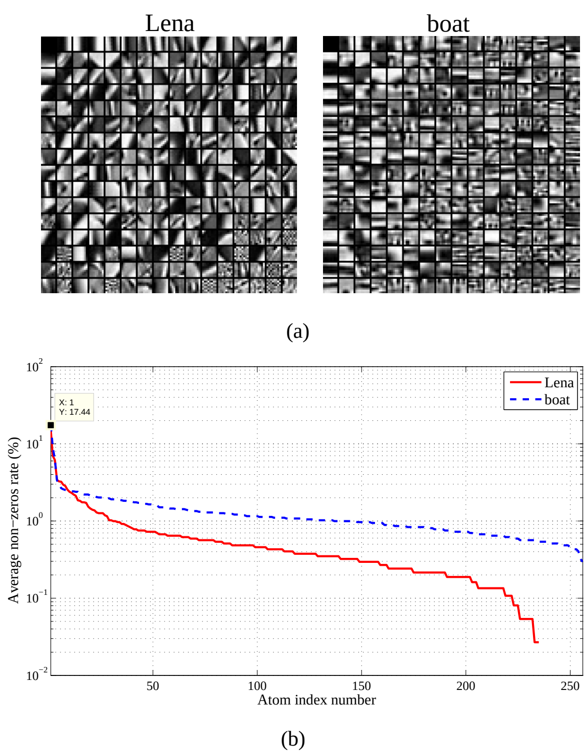


Fig. 5. (a) The obtained dictionaries as a result of applying the proposed method to the mixtures shown in Figure 4 (a). (b) The percentage of average number of non-zeros of sparse coefficients; equivalent to the occurrence rate of dictionary atoms sorted based on a descending order.

*Lena* and *texture*, respectively. The methods of FastICA[5] [32], JADE [2] and SMICA were also used in this experiment for comparison. We varied the noise standard deviation from $0$ to $20$ to investigate the performance of all these methods in dealing with different noise levels. Similar to [16] and [17], we calculated the *mixing matrix criterion* (also called Amari criterion [33]) as $\Gamma_{\mathbf{A}} = \|\mathbf{I} - \mathbf{P}\hat{\mathbf{A}}^{-1}\mathbf{A}\|_1$, where $\mathbf{P}$ is the scaling and permutation matrix.[6] The mixing matrix criterion curves are depicted in Figure 6 as a function of standard deviation. As it is seen from Figure 6, in low/moderate noise situations the best performance is achieved when both global and local dictionaries are used. However, the performance of the proposed method using only local, and both local and global dictionaries decreases in high noise. Moreover, the recovery results of JADE and FastICA are less accurate than those in the other methods. The performance of SMICA is somewhere between JADE's and that of the proposed method.

Next, in order to find an optimal $\lambda$, we investigated the effects of choosing different $\lambda$'s on the recovery quality. As expected and also shown in [1], selection of $\lambda$ is dependent on the noise standard deviation. However, our case is slightly different due to the source separation task.[7] In our simulations Gaussian noises with different power were added to the mixtures and the *mixing matrix criterion* was calculated while varying $\lambda$. Figure 7 represents the achieved results for this experiment. From this figure, it can be found that all the curves achieve nearly the same recovery quality for $\lambda\sigma = 30 \sim 40$. Therefore, $\lambda \approx 30/\sigma$ is a reasonable choice. Interestingly, these results and the range of appropriate choices for $\lambda$ are

[5]Available at: http://research.ics.tkk.fi/ica/fastica/

[6]The proposed method only causes signed permutations due to dealing with column-normalized mixing matrices.

[7]Indeed, we empirically observed better performance by starting with a higher value of $\sigma$ for solving (18) and decreasing it toward the true noise standard deviation, as the iterations proceed.

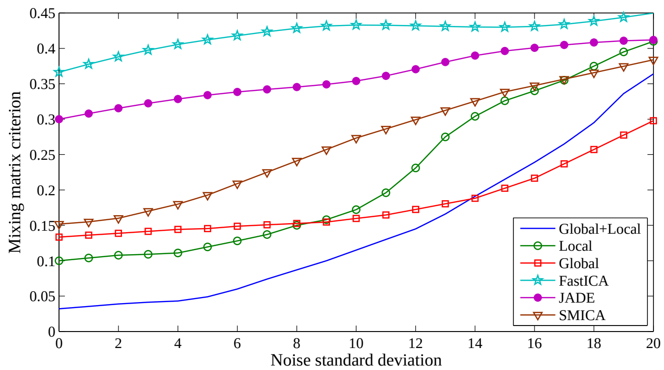


Fig. 6. The mixing matrix criterion as a function of noise standard deviation for different algorithms. Note that the "Global" method is actually equivalent to original MMCA in [16].

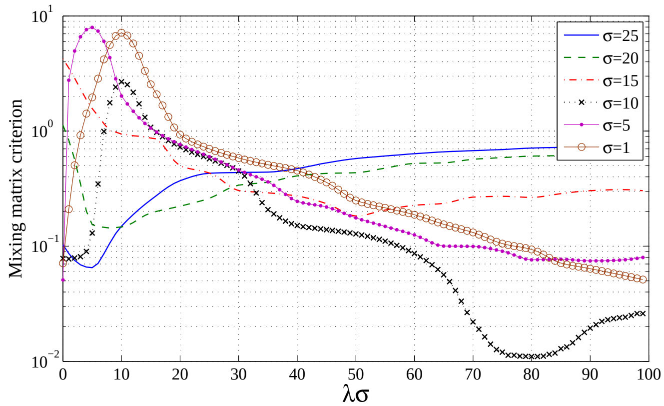


Fig. 7. Mixing matrix criterion as a function of $\lambda\sigma$.

very similar to what achieved in [1].

As already mentioned, it is expected that the computation time of the proposed method increases for large number of sources. A set of simulations were conducted to numerically demonstrate this effect and also to evaluate the effects of changing other dimensions such as patch size and dictionary size on the simulation time. In Table I we have given these results per one iteration of Algorithm 1, where the image patches had $50\%$ overlap. A desktop computer with a Core 2 Due CPU of 3 GHz and 2 GB of RAM was used for this experiment. It is interesting to note that as seen from Table I much of the computational burden is incurred by the dictionary update stage (using K-SVD). In addition, the computation of $\hat{\mathbf{x}}_{j:}$ is far less complicated than both "sparse coding" and "dictionary update", due to pixel-wise operation. This implies that further effort is required to speed up the dictionary learning part of the proposed algorithm.

## VII. Conclusions

In this paper the blind source separation problem has been addressed. The aim has been to take advantage of sparsifying dictionaries for this purpose. Unlike the existing sparsity-based methods, we assumed no prior knowledge about the underlying sparsity domain of the sources. Instead, we have proposed to fuse the learning of adaptive sparsifying dictionaries for each individual source into the separation process. Motivated by the idea of image denoising via a learned dictionary from the patches of the corrupted image in [1], we proposed a hierarchical approach for this purpose. Our simulation results on both noisy and noiseless mixtures have been encouraging and confirmed the effectiveness of the proposed approach. However, further work is required to speed up the algorithm and make it suitable for large-scale problems. In addition, the possibility of applying the proposed method to the *underdetermined* mixtures (less observations than sources) is of our future plans.

## Acknowledgment

The authors would like to acknowledge Professor Jean-François Cardoso for providing part of the MATLAB code for SMICA algorithm. They also would like to thank anonymous reviewers for their thorough remarks and fruitful suggestions.

## References


[1] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. on Image Processing*, vol. 15, no. 12, pp. 3736 –3745, December 2006.

[2] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. Wiley-Interscience, May 2001.

[3] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, October 1999. [Online]. Available: http://dx.doi.org/10.1038/44565

[4] K. Kayabol, E. E. Kuruoglu, and B. Sankur, "Bayesian separation of images modeled with MRFs using MCMC," *IEEE Trans. on Image Processing*, vol. 18, no. 5, pp. 982–994, 2009.

[5] J.-F. Cardoso, H. Snoussi, J. Delabrouille, and P. G., "Blind separation of noisy Gaussian stationary sources. Application to cosmic microwave background imaging," in *11th European Signal Processing Conference EUSIPCO*, Toulouse, France, September 2002, pp. 561–564.

[6] Y. Moudden, J.-F. Cardoso, J.-L. Starck, and J. Delabrouille, "Blind component separation in wavelet space: Application to CMB analysis," *EURASIP Journal on Applied Signal Processing*, vol. 2005, pp. 2437–2454, 2005.

[7] A. Tonazzini, L. Bedini, E. E. Kuruoglu, and E. Salerno, "Blind separation of auto-correlated images from noisy mixtures using MRF models," in *International Symposium on Independent Component Analysis and Blind Signal Separation – ICA'03*, Nara, Japan, April 2003, pp. 675–680.

[8] A. Tonazzini, L. Bedini, and E. Salerno, "A Markov model for blind image separation by a mean-field EM algorithm," *IEEE Trans. on Image Processing*, vol. 15, no. 2, pp. 473–482, 2006.

[9] H. Snoussi and A. Mohammad-Djafari, "Fast joint separation and segmentation of mixed images." *Journal of Electronic Imaging*, vol. 13, no. 2, pp. 349–361, 2004.

[10] E. Kuruoglu, A. Tonazzini, and L. Bianchi, "Source separation in noisy astrophysical images modelled by markov random fields," in *International Conference on Image Processing – ICIP '04*, vol. 4, October 2004, pp. 2701–2704.

[11] M. M. Ichir and A. Mohammad-Djafari, "Hidden Markov models for wavelet-based blind source separation," *IEEE Trans. on Image Processing*, vol. 15, no. 7, pp. 1887 –1899, July 2006.

[12] W. Addison and S. Roberts, "Blind source separation with non-stationary mixing using wavelets," in *International Conference on Independent Component Analysis and Signal Separation – ICA'06*, Charleston, SC, USA, 2006.

[13] A. M. Bronstein, M. M. Bronstein, M. Zibulevsky, and Y. Y. Zeevi, "Sparse ICA for blind separation of transmitted and reflected images," *International Journal of Imaging Systems and Technology*, vol. 15, no. 1, pp. 84–91, 2005.

[14] M. G. Jafari and M. D. Plumbley, "Separation of stereo speech signals based on a sparse dictionary algorithm," in *16th European Signal Processing Conference EUSIPCO*, Lausanne, Switzerland, August 2008, pp. 25–29.

TABLE I
THE DETAILED SIMULATION TIME OF ALGORITHM 1. "TOTAL" MEANS THE TOTAL TIME (IN SECOND) ELAPSED PER ONE ITERATION.

| Image size: $\sqrt{N} \times \sqrt{N}$ | $128 \times 128$ | | | | | | $256 \times 256$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mixing matrix size: $m$,$n$ | $4,2$ | $4,2$ | $8,4$ | $8,4$ | $16,8$ | $16,8$ | $4,2$ | $4,2$ | $8,4$ | $8,4$ | $16,8$ | $16,8$ |
| Patch size: $r$ | 16 | 64 | 16 | 64 | 16 | 64 | 16 | 64 | 16 | 64 | 16 | 64 |
| Dictionary atom size: $k$ | 64 | 256 | 64 | 256 | 64 | 256 | 64 | 256 | 64 | 256 | 64 | 256 |
| **Computation time (sec):** | | | | | | | | | | | | |
| Sparse coding (Batch-OMP) | 0.032 | 0.059 | 0.073 | 0.133 | 0.134 | 0.172 | 0.116 | 0.184 | 0.354 | 0.747 | 0.760 | 0.965 |
| Dictionary update | 0.366 | 1.229 | 0.724 | 2.314 | 1.450 | 4.391 | 0.565 | 1.594 | 1.532 | 4.786 | 2.772 | **7.675** |
| Residual calculation | 0.001 | 0.002 | 0.003 | 0.003 | 0.007 | 0.007 | 0.008 | 0.008 | 0.016 | 0.016 | 0.038 | 0.038 |
| $\hat{\mathbf{x}}_{j:}$ estimation | 0.081 | 0.023 | 0.161 | 0.047 | 0.324 | 0.094 | 0.325 | 0.93 | 0.654 | 0.196 | 1.326 | 0.395 |
| $\mathbf{a}_{:j}$ update and normalization | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.001 | 0.001 | 0.001 | 0.002 | 0.002 | 0.004 | 0.004 |
| Total | 0.540 | 1.333 | 1.091 | 2.547 | 2.229 | 4.815 | 1.260 | 1.961 | 3.104 | 5.966 | 6.300 | 9.802 |


[15] M. G. Jafari, M. D. Plumbley, and M. E. Davies, "Speech separation using an adaptive sparse dictionary algorithm," in *Joint Workshop on Hands-free Speech Communication and Microphone Arrays HSCMA*, Trento, Italy, May 2008, pp. 25–28.

[16] J. Bobin, Y. Moudden, J. Starck, and M. Elad, "Morphological diversity and source separation," *IEEE Signal Processing Letters*, vol. 13, no. 7, pp. 409 –412, July 2006.

[17] J. Bobin, J. Starck, J. Fadili, and Y. Moudden, "Sparsity and morphological diversity in blind source separation," *IEEE Trans. on Image Processing*, vol. 16, no. 11, pp. 2662 –2674, November 2007.

[18] J. Starck, M. Elad, and D. Donoho, "Redundant multiscale transforms and their application for morphological component separation," *Advances in Imaging and Electron Physics*, vol. 132, pp. 287 – 348, 2004.

[19] J.-L. Starck, M. Elad, and D. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Trans. on Image Processing*, vol. 14, no. 10, pp. 1570 –1582, October 2005.

[20] M. Elad, J. Starck, P. Querre, and D. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 340–358, November 2005.

[21] J. Bobin, J. Starck, J. Fadili, Y. Moudden, and D. Donoho, "Morphological component analysis: An adaptive thresholding strategy," *IEEE Trans. on Image Processing*, vol. 16, no. 11, pp. 2675 –2681, November 2007.

[22] G. Peyre, J. Fadili, and J. Starck, "Learning the morphological diversity," *SIAM Journal of Imaging Science*, vol. 3, no. 3, pp. 646–669, 2010.

[23] G. Peyre, J. Fadili, and J.-L. Starck, "Learning adapted dictionaries for geometry and texture separation," in *Storage and Retrieval for Image and Video Databases*, 2009.

[24] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010.

[25] N. Shoham and M. Elad, "Algorithms for signal separation exploiting sparse representations, with application to texture image separation," in *25th Convention of Electrical and Electronics Engineers in Israel, IEEEI 2008.*, December 2008, pp. 538 –542.

[26] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[27] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. on Information Theory*, vol. 53, no. 12, pp. 4655–4666, December 2007.

[28] I. Gorodnitsky and B. Rao, "Sparse signal reconstruction from limited data using focuss: a re-weighted minimum norm algorithm," *IEEE Trans. on Signal Processing*, vol. 45, no. 3, pp. 600 –616, March 1997.

[29] M. Yaghoobi, T. Blumensath, and M. Davies, "Parsimonious dictionary learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP*, April 2009, pp. 2869 –2872.

[30] S. Tjoa, M. Stamm, W. Lin, and K. Liu, "Harmonic variable-size dictionary learning for music source separation," in *IEEE International Conference on Acoustics Speech and Signal Processing ICASSP*, March 2010, pp. 413 –416.

[31] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," Technion, Tech. Rep., 2008.

[32] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. on Neural Networks*, vol. 10, no. 3, pp. 626 –634, May 1999.

[33] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Advances in Neural Information Processing Systems*. MIT Press, 1996, pp. 757–763.



**Vahid Abolghasemi** (S'08–M'12) received his Ph.D. degree in signal and image processing from University of Surrey, Guildford, UK, in 2011. He is currently with the School of Engineering and Design, Brunel University, Uxbridge, UK.

His main research interests include sparse signal and image analysis, dictionary learning, compressed sensing and blind source separation.

**Saideh Ferdowsi** (S'09) received both her B.Sc. and M.Sc. degrees in Electronic Engineering from Shahrood University of Technology, Shahrood, Iran, in 2005 and 2007 respectively. She is currently working towards her Ph.D. degree at the Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, UK.

Her main research interests include blind source separation, and biomedical signal and image processing.



**Saeid Sanei** (SM'05) received his Ph.D. degree in biomedical signal and image processing from Imperial College London, UK, in 1991.

His main area of research is signal processing and its application in biomedical engineering. He is an Associate Editor for IEEE Signal Processing Letters and EURASIP Journals. He is currently a Reader in Neurocomputing at the Faculty of Engineering and Physical Sciences, University of Surrey, UK.