# Morphological Component Analysis

J.-L. Starck [a,c], Y. Moudden [a], J. Bobin[a], M. Elad [b], and D.L. Donoho [c]

[a] DAPNIA/SEDI-SAP, Service d'Astrophysique,
CEA/Saclay, 91191 Gif sur Yvette, France
[b] Computer Science Department, Stanford University,
Stanford, CA, 94305, USA
[c] Statistic Department, Stanford University,
Stanford, CA, 94305, USA

## ABSTRACT

The Morphological Component Analysis (MCA) is a a new method which allows us to separate features contained in an image when these features present different morphological aspects. We show that MCA can be very useful for decomposing images into texture and piecewise smooth (cartoon) parts or for inpainting applications. We extend MCA to a multichannel MCA (MMCA) for analyzing multispectral data and present a range of examples which illustrates the results.

**Keywords:** Basis Pursuit Denoising, Total Variation, Sparse Representations, Piecewise Smooth, Texture, Wavelet, Local DCT, Ridgelet, Curvelet, ICA, Blind Source Separation.

## 1. INTRODUCTION

The task of decomposing signals into their building atoms is of great interest for many applications. In such problems a typical assumption is made that the given signal is a linear mixture of several source signals of more coherent origin. These problems have drawn a lot of research attention in last years. Independent Component Analysis (ICA) and sparsity methods are typically used for the separation of signal mixtures with varying degrees of success. A classical example is the cocktail party problem where a sound signal containing several concurrent speakers is to be decomposed into the separate speakers. In image processing, a parallel situation is encountered for example in cases of photographs containing transparent layers.

A dictionary $\mathcal{D}$ being defined as a collection of waveforms $(\varphi_\gamma)_{\gamma\in\Gamma}$, the general principle consists in representing a signal $s$ as a "sparse" linear combination of a small number of basis elements $\varphi_\gamma$ such that:

$$s = \sum_\gamma a_\gamma \varphi_\gamma \tag{1}$$

or as an approximate decomposition

$$s = \sum_{i=1}^{m} a_{\gamma_i} \varphi_{\gamma_i} + R^{(m)}. \tag{2}$$

Given $s$ and the dictionary atoms, an important question is the atom-decomposition problem, where we seek the representation coefficients $alpha_i$. While this is generally a hard task (combinatorial complexity), pursuit methods to approximate the desired coefficients are available.

The Matching Pursuit[1,2] method (MP) uses a greedy algorithm which adaptively refines the signal approximation with an iterative procedure:

- Set $s^0 = 0$ and $R^0 = 0$.
- Find the element $\alpha_k \varphi_{\gamma_k}$ which best correlates with the residual.

Further author information: Send correspondence to jstarck@cea.fr

- Update $s$ and $R$:

$$\begin{aligned} s^{k+1} &= s^k + \alpha_k \varphi_{\gamma_k} \\ R^{k+1} &= s - s^{k+1}. \end{aligned} \tag{3}$$

In the case of non orthogonal dictionaries, it has been shown[3] that MP may spend most of the time correcting mistakes made in the first few terms, and therefore is suboptimal in terms of sparsity.

The Basis Pursuit method[3] (BP) is a global procedure which synthesizes an approximation $\tilde{s}$ to $s$ by minimizing a functional of the type

$$\|s - \tilde{s}\|_{\ell_2}^2 + \lambda \cdot \|\alpha\|_{\ell_1} \text{ subject to } \tilde{s} = \Phi\alpha. \tag{4}$$

Among all possible solutions, the chosen one has the minimum $l^1$ norm. This choice of $l_1$ norm is very important. An $l_2$ norm, as used in the method of frames,[4] does not preserve sparsity.[3]

In many cases, BP or MP synthesis algorithms are computationally very expensive. We present in this paper an alternative to these approaches, the MCA method (Morphological Component Analysis) which can be seen as a kind of Basis Pursuit method in which i) our dictionary is a concatenation of sub-dictionaries which is associated to a transformation with fast forward and adjoint implementations, and ii) any kind of constraint can be easily imposed on the reconstructed components.

Section 2 presents the MCA approach. Section 3 and section 4 show respectively how MCA can be used for texture separation and inpainting. Two extensions to multichannel MCA are proposed in sections 5 and 6.

## 2. IMAGE DECOMPOSITION USING THE MCA APPROACH

### 2.1. Model Assumption

Assume that the data $s$ is a linear combination of $K$ parts, $s = \sum_{k=1}^{K} s_k$, where each $s_k$ represents a different type of signal to be decomposed. Our model assumes the following to hold true:

1. For every possible signal $s_k$, there exists a dictionary (which can be overcomplete), $\Phi_k \in \mathbf{M}^{N \times L_k}$ (where typically $L_A \gg N$) such that solving

$$\alpha_k^{opt} = \operatorname{Arg\,min}_{\alpha} \|\alpha\|_0 \text{ subject to: } s_A = \Phi_A \alpha \tag{5}$$

   leads to a very sparse solution (i.e. $\|\alpha_k^{opt}\|_0$ is very small). The definition in the above equation is essentially the overcomplete transform of $s_k$, yielding a representation $\alpha_k$.

2. For every possible signal $s_l$, solving for $k \neq l$

$$\alpha_l^{opt} = \operatorname{Arg\,min}_{\alpha} \|\alpha\|_0 \text{ subject to: } s_l = \Phi_k \alpha \tag{6}$$

   leads to a very **non-sparse** solution. This requirement suggests that the dictionary $\Phi_k$ is distinguishing between the different types of signals to be separated.

Thus, the dictionaries $\Phi_k$ play a role of discriminants between the different content types.

Finally, we consider only dictionaries $\Phi_k$ which have a fast transformation $\mathbf{T}_k$ ($\alpha_k = \mathbf{T}_k s_k$) and reconstruction $\mathbf{R}_k$ ($s_k = \mathbf{R}_k \alpha_k$).

### 2.2. The MCA concept

For an arbitrary signal $s$ containing $K$ layers as a linear combination, we propose to seek the sparsest of all representations over the augmented dictionary containing all $\Phi_k$. Thus we need to solve

$$\begin{aligned} \{\alpha_1^{opt}, \ldots, \alpha_K^{opt}\} &= \underset{\{\alpha_1, \ldots, \alpha_K\}}{\text{Arg min}} \sum_{k=1}^{K} \|\alpha_k\|_0 \\ &\text{subject to: } s = \sum_{k=1}^{K} \Phi_k \alpha_k. \end{aligned} \tag{7}$$

This optimization task is likely to lead to a successful separation of the signal content, based on the assumptions made earlier about $\Phi_k$ being very efficient in representing one phenomenon and being highly non-effective in representing the other signal types.

While sensible from the point of view of the desired solution, the problem formulated in Equation (7) is non-convex and hard to solve. Its complexity grows exponentially with the number of columns in the overall dictionary. The Basis Pursuit (BP) method[3] suggests the replacement of the $\ell^0$-norm with an $\ell^1$-norm, thus leading to a solvable optimization problem (Linear Programming) of the form

$$\begin{aligned} \{\alpha_1^{opt}, \ldots, \alpha_K^{opt}\} &= \underset{\{\alpha_1, \ldots, \alpha_K\}}{\text{Arg min}} \sum_{k=1}^{K} \|\alpha_k\|_1 \\ &\text{subject to: } s = \sum_{k=1}^{K} \Phi_k \alpha_k. \end{aligned} \tag{8}$$

Interestingly, recent work has shown that for sparse enough solutions, the BP simpler form is accurate, also leading to the sparsest of all representations.[5–8]

A solution for the first problem could be obtained by relaxing the constraint in Equation (8) to become an approximate one. Thus, in this new form, we seek to solve

$$\{\alpha_1^{opt}, \ldots, \alpha_K^{opt}\} = \underset{\{\alpha_1, \ldots, \alpha_K\}}{\text{Arg min}} \sum_{k=1}^{K} \|\alpha_k\|_1 + \lambda \left\| s - \sum_{k=1}^{K} \Phi_k \alpha_k \right\|_2^2. \tag{9}$$

We should note here that the choice of $\ell^2$ as the error norm is intimately related to the assumption that the residual behaves like a white zero-mean Gaussian noise. Other norms can be similarly introduced to account for different noise models, such as Laplacian ($\ell^1$), uniformly distributed noise ($\ell^\infty$), and others.

Another complicating factor is the length $L$ of the representation vector $\alpha_{all}$. If for example $L = 100N$ (implying a redundancy of factor 100), it means that storing and manipulating the solution of this problem requires a memory of 100 the input data size. Instead of solving this optimization problem, finding the representation vectors $\{\alpha_1^{opt}, \ldots, \alpha_K^{opt}\}$, let us reformulate the problem so as to get the $K$ signal types, $\{s_1, \ldots, s_K\}$, as our unknowns. This way, if we return to the example mentioned above, we seek $K$ images rather than 100. The functional to minimize is now:

$$\{s_1^{opt}, \ldots, s_K^{opt}\} = \underset{\{s_1, \ldots, s_K\}}{\text{Arg min}} \sum_{k=1}^{K} \|\mathbf{T}_k s_k\|_1 + \lambda \left\| s - \sum_{k=1}^{K} s_k \right\|_2^2. \tag{10}$$

and the unknowns become images, rather then representation coefficients. For this problem structure there exists a fast numerical solver called the *Block-Coordinate Relaxation Method*, based on the shrinkage method.[9] This solver requires *only* the use of matrix-vector multiplications with the unitary transforms and their inverses. See[10] for more details. We note that the same algorithm can be applied with non-unitary transforms although

theoretical validation may require more challenging analysis than in the block-unitary case. In practice, block coordinate methods work well in the non-unitary cases we have explored.

Finally one important aspect of working on the signals rather than on the coefficients is the possibility to add constraints on each individual signal $s_k$. The MCA method consists in minimizing

$$\{s_1^{opt}, \ldots, s_K^{opt}\} = \underset{\{s_1, \ldots, s_K\}}{\text{Arg min}} \sum_{k=1}^{K} \|T_k s_k\|_1 + \lambda \left\| s - \sum_{k=1}^{K} s_k \right\|_2^2 + \sum_{k=1}^{K} \gamma_k \mathcal{C}_k(s_k). \quad (11)$$

where $\mathcal{C}_k$ implements constraints on component $s_k$.

### 2.3. Algorithm

We have chosen an approximation to our true minimization task, and with it managed to get a simplified optimization problem, for which an effective algorithm can be proposed. The algorithm we use is based on the Block-Coordinate-Relaxation method,[10] with some required changes due to the non-unitary transforms involved. The algorithm is given below:

1. Initialize $L_{\max}$, number of iterations, and threshold $\delta = \lambda \cdot L_{\max}$.
2. Perform $J$ times:
    3. Perform $K$ times:
        Update of $s_k$ assuming all $s_l, l \neq k$, are fixed:
        - Calculate the residual $r = s - \sum_{l=1, l\neq k}^{K} s_l$
        - Calculate the transform $\mathbf{T}_k$ of $s_k + r$ and obtain $\alpha_k = \mathbf{T}_k(s_k + r)$.
        - Soft threshold the coefficient $\alpha_k$ with the $\delta$ threshold and obtain $\hat{\alpha}_k$.
        - Reconstruct $s_k$ by $s_k = \mathbf{R}_k \hat{\alpha}_k$.
        - Apply the constraint correction by $s_k = s_k - \mu\gamma \frac{\partial \mathcal{C}_k\{s_k\}}{\partial s_k}$.
        - The parameter $\mu$ is chosen either by a line-search minimizing the overall
3. Update the threshold by $\delta = \delta - \lambda$.
4. If $\delta > \lambda$, return to Step 2. Else, finish.

The numerical algorithm for minimizing (11). *.

In the above algorithm, soft thresholdong is used due to our formulation of the $\ell^1$ sparsity penalty term. However, as we have explained earlier, the $\ell^1$ expression is merely a good approximation for the desired $\ell^0$ one, and thus, replacing the soft by a hard threshold towards the end of the iterative process may lead to better results.

We chose this numerical scheme over the Basis Pursuit interior-point approach in ,[3] because it presents two major advantages:

- We do not need to keep all the transformations in memory. This is particularly important when we use redundant transformations such the un-decimated wavelet transform or the curvelet transform.
- We can add different constraints on the components. As we shall see next, Total-Variation on some of the content types may support the separation task, and other constraints, such as positivity, can easily be added as well.

*Notice that in turning from the formulation (11) to the algorithm described here, we have changed the role of $\lambda$. In the algorithm it is used as a weight that multiplies the $\ell^1$-norm terms. This change was made to better fit the soft-thresholding description, and it has no impact on the way the problem formulation acts.

**Noise Consideration**

The case of noisy data can be easily considered in our framework, and merged into the algorithm such that we get a separation between components and an additive noise $n$: $s = \sum_{k=1}^{K} s_k + n$. We can normalize each transform $\mathbf{T}_k$ such that for a given noise realization $n$ with zero-mean and a unit standard deviation, $\alpha_k = \mathbf{T}_k n$ has also a standard deviation equal to 1. Then, only the last step of the algorithm changes. By replacing the stopping criterion $\delta > \lambda$ by $\delta > a\sigma$, where $\sigma$ is the noise standard deviation and $a \approx 3,4$. This ensures that coefficients with an absolute value lower than $a\sigma$ will never be taken into account.

## 2.4. Example

Figure 1 illustrates the separation results in the case where the input image ($256 \times 256$) contains only lines and isotropic Gaussians. In this experiment, we have initialized $L_{\max}$ to 20, and $\delta$ to 2 (10 iterations). Two transform operators were used, the isotropic undecimated wavelet transform and the ridgelet transform. The first is well adapted to the detection of the isotropic Gaussians due to the isotropy of the wavelet function,[11] while the second is optimal to represent lines.[12] Figure 1 represents respectively the original image, the reconstructed image from the à trous wavelet coefficient, and the reconstructed image from the ridgelet coefficient. The addition of both reconstructed images reproduces the original one.

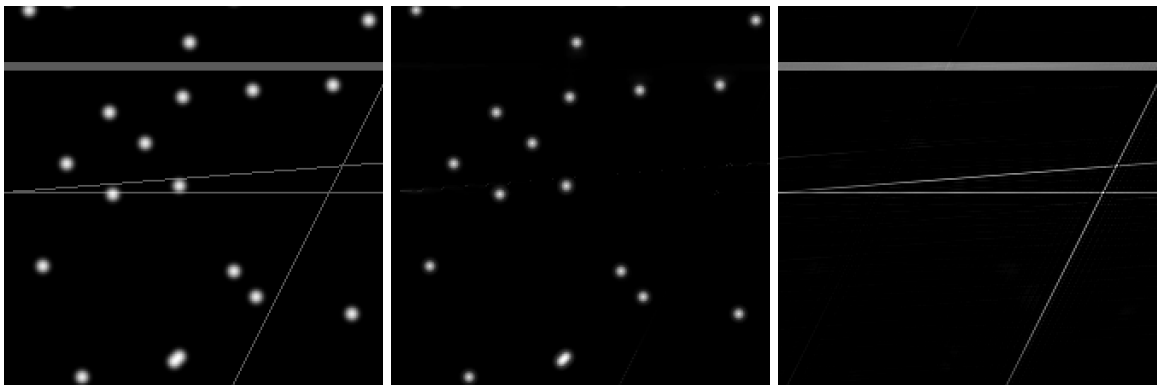


**Figure 1.** Left, original image containing lines and Gaussians. Middle, reconstructed image for the à trous wavelet coefficient, bottom right, reconstructed image from the Ridgelet coefficients.

The above experiment is synthetic and through it we validate the proper behavior of the numerical scheme proposed. While being synthetic, this experiment has also high relevance for astronomical data processing where stars look like Gaussians and where images may also contain anisotropic features (dust emission, supernovae remnants , filaments, ...). Separation of these components is very important for the analysis of this type of images.

# 3. TEXTURE SEPARATION USING MCA

An interesting and complicated image content separation problem is the one targeting decomposition of an image to texture and piece-wise-smooth (cartoon) parts. Such separation finds applications in image coding, and in image analysis and synthesis (see for example [13]).

A theoretic characterization of textures proposed recently by Meyer[14] was used by Vese and Osher ,[15] and Aujol et al.[16] for the design of such image separation algorithms, and these pioneering contributions awaken this application field. The approach advocated by Vese and Osher [15] is built on variational grounds, extending the notion of Total-Variation.[17]

Here we demonstrate that the is capable of separating these image content types, and as such poses an alternative method to the variational one mentioned above. More on this approach can be found in .[18,19]

For the texture description, the DCT seems to have good properties due to the natural periodicity. If the texture is not homogeneous, a local DCT should be preferred. Characterizing the cartoon part of the image could be done in various ways, depending on the image content. For images containing lines of a fixed size, the local ridgelet transform will be a good dictionary candidate. More generally the curvelet transform represents well edges in images, and could be a good candidate as well. In our experiments, we have chosen images with edges, and decided to apply the texture/signal separation using the DCT and the curvelet transform.

Assume hereafter that we use the DCT for the texture - denoted $\mathbf{D}$ - the curvelet transform for the natural scene part, denote $\mathbf{C}$. Returning to the separation process as posed earlier, we have two unknowns - $s_d$ and $s_c$ ($K = 2$)- the texture and the piecewise smooth images. The optimization problem to be solved is[20]:

$$\min_{\{s_d,\ s_c\}} \quad \|\mathbf{D}s_d\|_1 + \|\{\mathbf{C}s_c\|_1 + \lambda \|s - s_d - s_c\|_2^2 + \gamma TV\{s_c\}\,. \tag{12}$$

In this optimization problem we support the choice of the cartoon dictionary by adding another penalty term based on the Total-Variation on the cartoon image part.

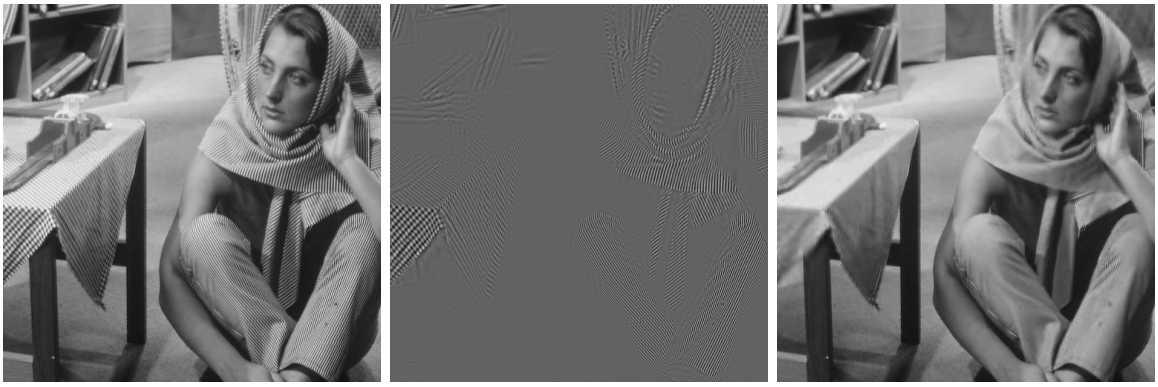


**Figure 2.** Left, the original `Barbara` image; midlle, the separated texture, and right, the separated cartoon.

Figure 2 illustrates the layer separation result for the `Barbara` image, as obtained with the algorithm described above. Many more such results are given in .[18,19] This separation was obtained using the curvelet transform with five resolution levels of the curvelet transform $\mathbf{C}$, and 50% overlapping discrete cosine transform $\mathbf{D}$ with a block size $32 \times 32$.

## 4. IMAGE INPAINTING USING MCA

Filling–in 'holes' in images is an interesting and important inverse problem with many applications. Removal of scratches in old photos, removal of overlayed text or graphics, filling–in missing blocks in unreliably transmitted images, scaling–up images, predicting values in images for better compression, and more, are all manifestations of the above problem. In recent years this topic has attracted much interest, and many contributions have been proposed for the solution of this interpolation task. A common feature of these many techniques is the understanding that classical interpolation methods (such as polynomial-based approaches) are not satisfactory; indeed nonlinear strategies and local adaptivity seem crucial. Among the numerous approaches to fill in holes in images, variational methods are very attractive; these were pioneered by Guillermo Sapiro and his collaborators ,[21–23] and followed by .[24] These techniques were coined *Inpainting* as a reminder of the recovery process museum experts use for old and deteriorated artwork.

Based on MCA, an inpainting algorithm has been proposed [20] which is capable of filling in holes in either texture or cartoon content, or any combinations thereof. This new algorithm extends the sparsity–seeking layer separation method of[18,19] mentioned above. In effect, missing pixels fit naturally into the layer-separation framework. As a result, layer separation and denoising of the image are integral by-products of the inpainting process.

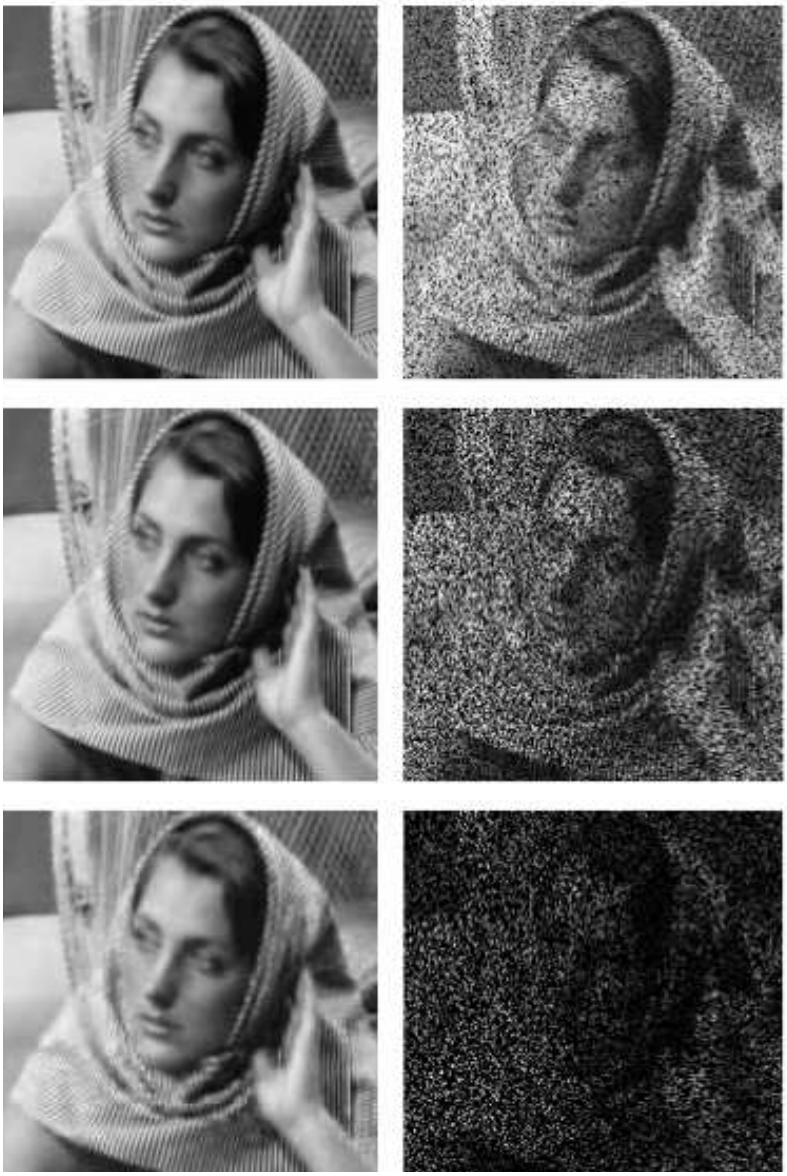


**Figure 3.** Three `Barbara` images with 20%, 50%, and 80% missing pixels (right). The results of the MCA inpainting are given on the left.

Assume that the missing pixels are indicated by a diagonal 'mask' matrix $\mathbf{M} \in \mathcal{M}^{N\times N}$. The main diagonal of $\mathbf{M}$ encodes the pixel status, namely '1' for an existing pixel and '0' for a missing one. Then model (11) can be modified to incorporate this mask as in

$$\min_{\{s_d,\ s_c\}} \quad \|\mathbf{D}s_d\|_1 + \|\{\mathbf{C}s_c\|_1 + \lambda \|M\left[s - s_d - s_c\right]\|_2^2 + \gamma TV\left\{s_c\right\}. \tag{13}$$

Clearly, replacing the mask matrix by the identity operator leads to the very same algorithm as proposed in[18, 19] for the task of image decomposition. Thus, this algorithm is a simple modification of the separation one proposed earlier. In the algorithm, the only required modification consists in multiplying the residual by $\mathbf{M}$ after each residual estimation.

The rationale behind the way the mask is taken into account here is the following: suppose that after several rounds we have a rough approximation of $s_d$ and $s_c$. In order to update $s_d$ we assume that $s_c$ is fixed and compute the residual image $r = \mathbf{M}(s - s_c - s_d)$. In existing pixels (where the mask value is '1') this residual has a content that can be attributed to texture, cartoon, and/or noise content. On the missing pixels (where the mask is '0') the residual value is forced to zero by the multiplication with the mask. Thus, the image $r + s_d$ does not contain holes. An analysis of this image – transforming it to curvelet coefficients, zeroing small entries, and reconstructing it – is able to absorb some of the cartoon content that exists in $r$. This way the updated $s_d$ takes some of the cartoon content that exists in the residual, and the new residual image energy becomes smaller. More details and experiments can be found in.[20]

**Experiment - Random Mask**: Figure 3 presents the `Barbara` image and its filled-in results for three random patterns of 20%, 50%, and 80% missing pixels. The unstructured form of the mask makes the reconstruction task easier. These results are tightly related to the removal of salt-and-pepper noise in images. As before, the MCA-inpainting method applied here used Wavelet and Wavelet Packets to represent the cartoon and the texture respectively, and again, the results look natural and artifact-free.

## 5. MULTICHANNEL MCA

### 5.1. Blind Source Separation

Blind Source Separation (BSS) is a problem that occurs in multi-dimensional data processing. The overall goal is to recover unobserved signals, images or *sources* $S$ from mixtures $X$ of these sources observed typically at the output of an array of sensors. The simplest mixture model would take the form:

$$X = AS \tag{14}$$

where $X$, $S$ and $A$ are matrices of respective sizes $n_c \times n$, $n_s \times n$ and $n_c \times n_s$. Multiplying $S$ by $A$ linearly mixes the $n_s$ sources into $n_c$ observed processes. Independent Component Analysis methods were developed to solve the BSS problem, *i.e.* estimate $A$ and $S$, relying mostly on the statistical independence of the source processes. Although independence is a strong assumption, it is plausible in many practical cases.

Algorithms for blind component separation and mixing matrix estimation depend on the model used for the probability distributions of the sources.[25] In a first set of techniques, source separation is achieved in a noise-less setting, based on the non-Gaussianity of all but possibly one of the components. Most mainstream ICA techniques belong to this category : Jade,[26] FastICA, Infomax.[27]

In a second set of blind techniques, the components are modeled as Gaussian processes, either stationary or non stationary and, in a given representation, separation requires that the sources have diverse, *i.e.* non proportional, variance profiles. The Spectral Matching ICA method (SMICA) [28] considers in this sense the case of mixed stationary Gaussian components and goes beyond the above model (Eq. 14) by taking into account additive *instrumental* noise $N$:

$$X = AS + N \tag{15}$$

Moving to a Fourier representation, the point of SMICA is that colored components can be separated based on the diversity of their power spectra.

### 5.2. MMCA

Many papers have recently proposed that sparsity could be used to build alternative source separation methods.[29,30] Following this idea, we extended MCA to deal with multichannel data (MMCA) leading to a powerful separation algorithm.

As before, we assume that each source $s_k$ is well represented (i.e. sparsified) by a given transform, but now, the observed data $X$ are no longer the sum of sources, but a set of $n_c$ linear combinations of the $n_s$ sources: $X_l = \sum_{k=1}^{n_s} A_{k,l} s_k$, where $l = 1 \dots n_c$, $A$ is the mixing matrix and, here, $s_k$ is the $1 \times n$ array of the $k^{\text{th}}$ source samples. The solution $S = \{s_1, \dots, s_{n_s}\}$ and $A$ is obtained as

$$\{s_1^{opt}, \dots, \ s_{n_s}^{opt}, A^{opt}\} \quad = \quad \underset{\{s_1, \dots, \ s_{n_s}\}}{\text{Arg min}} \ \sum_{k=1}^{n_s} \frac{1}{\lambda_k} \|s_k \mathbf{T}_k\|_1 + \ \|X - AS\|_2^2 \tag{16}$$

Unfortunately, this criterion suffers from several drawbacks and particularly from an indeterminacy attached to the model structure. For instance, assuming that the simple scale transform occurs: $A \leftarrow \rho A$ then an inverse scaling of the source matrix such that $S \leftarrow \frac{1}{\rho} S$ leaves the error minimization constraint unchanged whereas the sparsity measure is deeply altered by the same scale factor $\frac{1}{\rho}$. Consequently, the minimization of this criterion will probably lead to trivial solutions : $A \to \infty$ and $S \to 0$ (the sparsity term can be minimized as desired as long as $\rho$ tends to $+\infty$).

The multichannel extension of the MCA algorithm must therefore take into account this lack of scale invariance by introducing artificially this essential property. Practically, lack of scale invariance can be solved by normalizing each column of $A$ at each iteration ($a^{k^+} \leftarrow \frac{a^{k^-}}{\|a^{k^-}\|_2}$) and propagating the scale factor $\|a^{k^-}\|_2$ to the corresponding source $s_k$ and threshold $\lambda_k$ such that ${s_k}^+ \leftarrow \|a^{k^-}\|_2 {s_k}^-$ and ${\lambda_k}^+ \leftarrow \|a^{k^-}\|_2 {\lambda_k}^-$.

The model structure can be simplified by decomposing the product $AS$ by a sum of elementary products of directions and sources as follows:

$$X = \sum_{k=1,\dots,n_s} a^k s_k \tag{17}$$

Introducing the k-th residual, $D_k = X - \sum_{j \neq k} a^j s_j$ (corresponding to the part of the data unexplained by the other couples $\{a^j, s_j\}_{j \neq k}$), the minimization of the whole criterion 16 is equivalent to jointly minimizing the set of elementary criteria :

$$\{s_k^{opt}, \ {a^k}^{opt}\} \quad = \quad \underset{\{s_k, \ a^k\}}{\text{Arg min}} \ \|s_k \mathbf{T}_k\|_1 + \lambda_k \left\|D_k - a^k s_k\right\|_2^2 \tag{18}$$

Estimating jointly couples of directions and sources is also widely interesting as it avoids the computation of matrix inverses given that the vector products ${a^k}^T a^k$ or $s_k {s_k}^T$ found in the algorithm are just scalars. If we also assume that the covariance matrix $\Gamma_n$ of noise is known, the criterion is now written as below:

$$\{s_k^{opt}, \ {a^k}^{opt}\} \quad = \quad \underset{\{s_k, \ a^k\}}{\text{Arg min}} \ \|s_k \mathbf{T}_k\|_1 + \lambda_k Trace\{(D_k - a^k s_k)\Gamma_n^{-1}(D_k - a^k s_k)^T\} \tag{19}$$

Zeroing the gradient with respect to $s_k$ and $a^k$ of this criterion leads to the following coupled equations:

$$\begin{cases} s_k \quad = \quad \frac{1}{{a^k}^T \Gamma_n^{-1} a^k} \left( {a^k}^T \Gamma_n^{-1} D_k - \frac{1}{2\lambda_k} \text{Sign}(s_k \mathbf{T}_k) \mathbf{T}_k^T \right) \\ a^k \quad = \quad \frac{1}{s_k {s_k}^T} D_k {s_k}^T \end{cases} \tag{20}$$

It appears that, considering a fixed $a^k$, the source process $s_k$ is estimated by soft-thresholding a transformed "coarse version" $\widetilde{s_k} = \frac{1}{{a^k}^T \Gamma_n^{-1} a^k} {a^k}^T \Gamma_n^{-1} D_k$ with threshold $\frac{1}{\lambda_k}$. Then, considering a fixed $s_k$, the update on $a_k$ follows from a simple least squares linear regression. In comparison to the algorithm in[29] which uses a single sparsifying transform and a quadratic programming technique, our method considers more than just one transform and a shrinkage-based optimization. The MMCA algorithm is then given below :

1. Initialize $L_{\max}$, number of iterations, and threshold $\forall k\ \delta_k = \lambda_k \cdot L_{\max}$.
2. Perform $J$ times:
   3. For $k = 1, \ldots, n_s$:
      Normalization and propagation for scale invariance:
      $n_{ak} = \|a^k\|_2$, $a^k = \frac{a^k}{n_{ak}}$, $s_k = n_{ak} s_k$ and $\delta_k = n_{ak} \delta_k$
      Estimation of $s_k$ assuming all $s_l, l \neq k$ and $a^l$ are fixed:
      - Calculate the residual $D_k = X - \sum_{l=1, l \neq k} a^l s_l$
      - Projection of the residual $\tilde{s}_k = \frac{1}{{a^k}^T \Gamma_n^{-1} a^k} {a^k}^T \Gamma_n^{-1} D_k$
      - Calculate $\alpha_k = \tilde{s}_k \mathbf{T}_k$.
      - Soft threshold the coefficients $\alpha_k$ with the $\delta_k$ threshold and obtain $\hat{\alpha}_k$.
      - Reconstruct $s_k$ by $s_k = \hat{\alpha}_k R_k$.
      Estimation of $a^k$ assuming all $s_l$ and $a^l_{l \neq k}$ , are fixed:
      - Estimation of $a^k$ by $a^k = \frac{1}{s_k {s_k}^T} D_k {s_k}^T$.
3. Update the threshold by $\delta_k = \delta_k - \lambda_k$.
4. If $\delta_k > \lambda_k$, return to Step 2. Else, finish.

The numerical algorithm for minimizing (18).

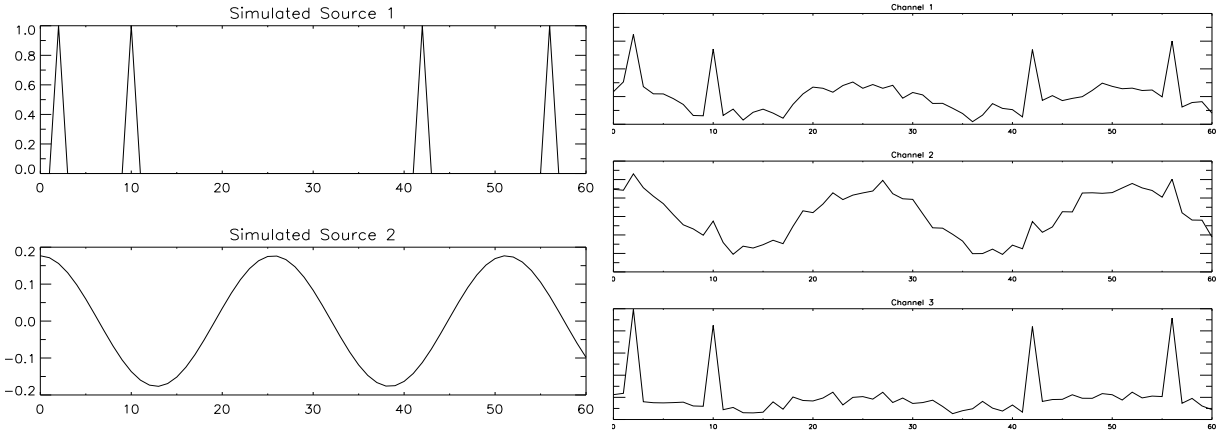



**Figure 4.** Simulated data. Left, the original source and right, the three observed channels. Each channel is a linear combination of the two sources plus some Gaussian noise.

Figure 4 shows on its left two simulated sources, the first containing four bumps and the second a sine, and on the right, three simulated observed channels. Gaussian noise was added to each channel and the resulting mixtures are shown on the right. Figure 5 shows the reconstructed sources using two methods, JADE (see [26]) on the left, and MMCA on the right.

This algorithm brings a very strong and robust solution as long as the MCA hypothesis is verified (sources are sparsified in different bases) i.e. for morphologically diverse sources. Actually source separation is merely a

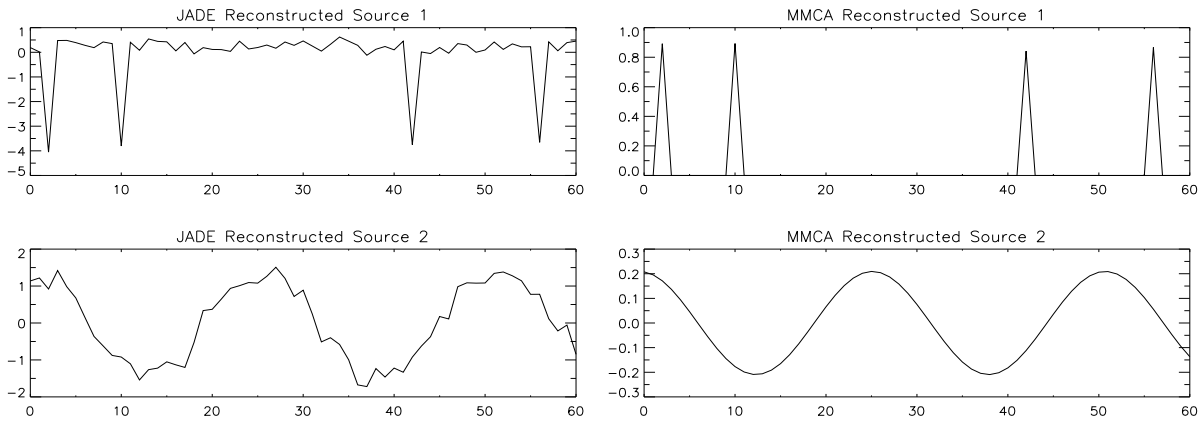



**Figure 5.** Left, reconstructed sources using JADE. Right, reconstructed sources using MMCA.

question of diversity. On the one hand standard ICA methods aim at enforcing the independence of the estimated sources and suffer from a certain lack of robustness in the presence of noise. On the other hand MMCA performs well even in the presence of noise for morphologically different sources. Next we describe an extension of MMCA coined ICA-MCA which aims at getting the most of both ICA and MMCA.

## 6. INDEPENDENT AND SPARSE SOLUTION: THE ICA-MCA METHOD

In this section we introduce a brand new method coined ICA-MCA devised to separate independent sources in the presence of noise, considering a more general model for the sources according to which each source $s_i$ $(i = 1 \dots n)$ is a sum of $K_i$ components:

$$s_i = \sum_{k=1}^{K_i} c_{i,k} \tag{21}$$

And the observed data are:

$$X_l = \sum_{i=1}^{n} A_{i,l} s_i = \sum_{i=1}^{n_s} A_{i,l} \sum_{k=1}^{K_i} c_{i,k} \tag{22}$$

The previous MMCA model corresponds to the case where $K_1 = \ldots = K_n = 1$. As before, we assume that each component $c_{i,k}$ is sparsified using a given transform $\mathbf{T}_{i,k}$. $\mathbf{T}_{i,k}$ is the transform associated to the $k$-th component of the $i$th source. Nothing here forces the transforms related to a given source to be different from those related to another one. For instance, we can have $\mathbf{T}_{1,k} = \ldots = \mathbf{T}_{n,k}$, in the case where the two different sources contain the same morphological components.

We make at this point the additional assumption that A is square and of full rank. It seems easier then to estimate a demixing matrix $B$ rather than a mixing matrix. This way, we avoid the issues raised by the bilinearity $AS$. Finally we seek to solve the following minimization problem:

$$\{c_{1,1}^{opt}, \ldots, c_{1,K_1}^{opt}, \ldots, c_{n_s,1}^{opt}, \ldots, c_{n_s,K_{n_s}}^{opt}, B^{opt}\} = \underset{\{s_1,\ldots,\ s_{n_s}\}}{\text{Arg min}} \sum_{i=1}^{n_s} \sum_{k=1}^{K_i} \frac{1}{\lambda_k} \|c_{i,k} \mathbf{T}_{i,k}\|_1 + \|BX - S\|_2^2 \tag{23}$$

ICA-MCA takes advantage of both ICA, which enforces the statistical indepedance of the estimated sources, and MCA, which gives more contrast between the sources as it extracts the essence of each one. The ICA-MCA algorithm is given below :

1. Perform $J$ times.

   Data projection:

   – Project the data $\tilde{S} = \tilde{B}X$

   MCA Decomposision :

   – $\forall\ k = 1 \ldots n_s$ calculate $\alpha_k = s_k \mathbf{T}_k$.

   Update of the demixing matrix using an ICA-kind update rule assuming that all the $\{\alpha_k\}_{k=1,\ldots,n_s}$ are fixed.

The numerical algorithm for minimizing (23).

In a few words, the ICA-MCA is divided into three steps:

1. Project the data to get a "coarse estimation" of the source: this minimizes the quadratic part of the criterion described above.
2. Apply MCA to each "coarse" source in ordre to achieve a sparse decomposition of each estimated source, thus eliminating all kind of redundancies.
3. Update the estimated demixing matrix $B$ using an ICA-update rule to enforce the statistical independence of the sources as in [27] where the update of the demixing matrix is achieved by a few steps of the natural gradient of the likelihood of $B$.

**Experiment -** : Figure 6 shows two simulated observed channels (a gaussian noise of variance 0.03 was added to each channel), obtained from a mixture of two sources shown in Figure 7 top left and top right. Both sources contain sinusoids and bumps. Figure 7 middle left and right shows the ICA-MCA reconstructed sources and Figure 7 bottom left and right shows the reconstructed sources using the JADE method. In this experiment, we have initialized $J$ to 20 iterations. The sources were decomposed on the Dirac basis ($T_1$ was the identity matrix) and using the DCT. One can notice that the low amplitude sinusoid of source 1 is not detected by JADE. The ICA-MCA method achieves such a detection quite well.

Figure 8 left shows the simulated ($\triangle$) and reconstructed (using ICA-MCA) coefficients of source 1. On the right, the simulated ($\triangle$) and reconstructed (using ICA-MCA) coefficients of source 2. Each decomposition is achieved on the whole dictionnary : samples 0 to 149 correspond to the Dirac basis and samples 150 to 299 correspond to the DCT basis. Due to the scale indeterminacy discussed previously, only their relative amplitudes of the coefficients are significant.

We further assess the quality of the separation by computing the angular error between the original demixing matrix $B^{(0)}$ and the estimated one $B$. Estimating the first direction with JADE leads to an angular error of 1.13 and 0.73 degrees for the first and the second directions respectively. ICA-MCA leads to an angular error of 0.13 and 0.45 degrees.

## REFERENCES


1. S. Mallat and Z. Zhang, "Atomic decomposition by basis pursuit," *IEEE Transactions on Signal Processing* **41**, pp. 3397–3415, 1993.
2. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1998.
3. S. Chen, D. Donoho, and M. Saunder, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing* **20**, pp. 33–61, 1998.
4. I. Daubechies, "Time-frequency localization operators: A geometric phase space approach," *IEEE Transactions on Information Theory* **34**, pp. 605–612, 1988.
5. D. Donoho and X. Huo, "Uncertainty Principles and Ideal Atomic Decomposition," *IEEE Transactions on Information Theory* **47**(7), pp. 2845–2862, 2001.
6. D. L. Donoho and M. Elad, "Maximal sparsity representation via $l_1$ minimization," *the Proc. Nat. Aca. Sci.* **100**, pp. 2197–2202, 2003.

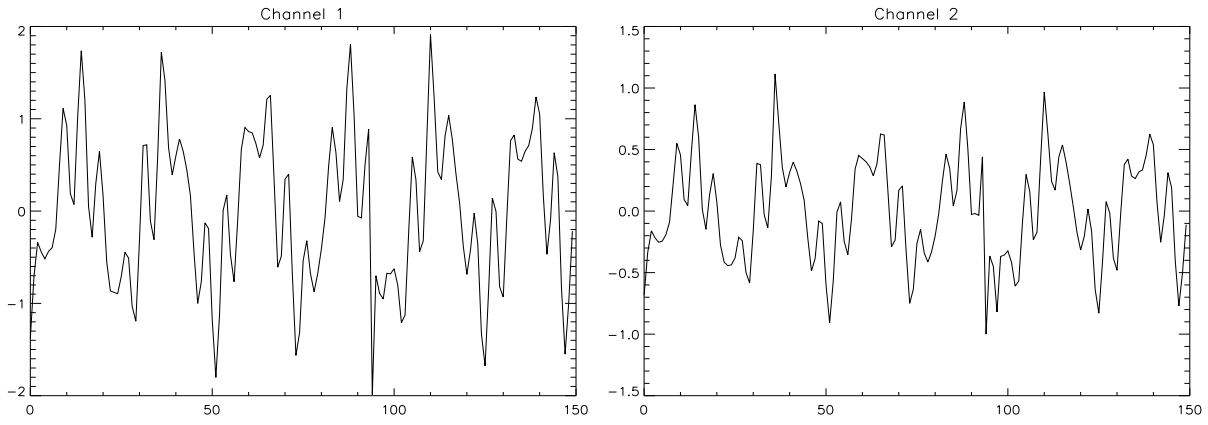
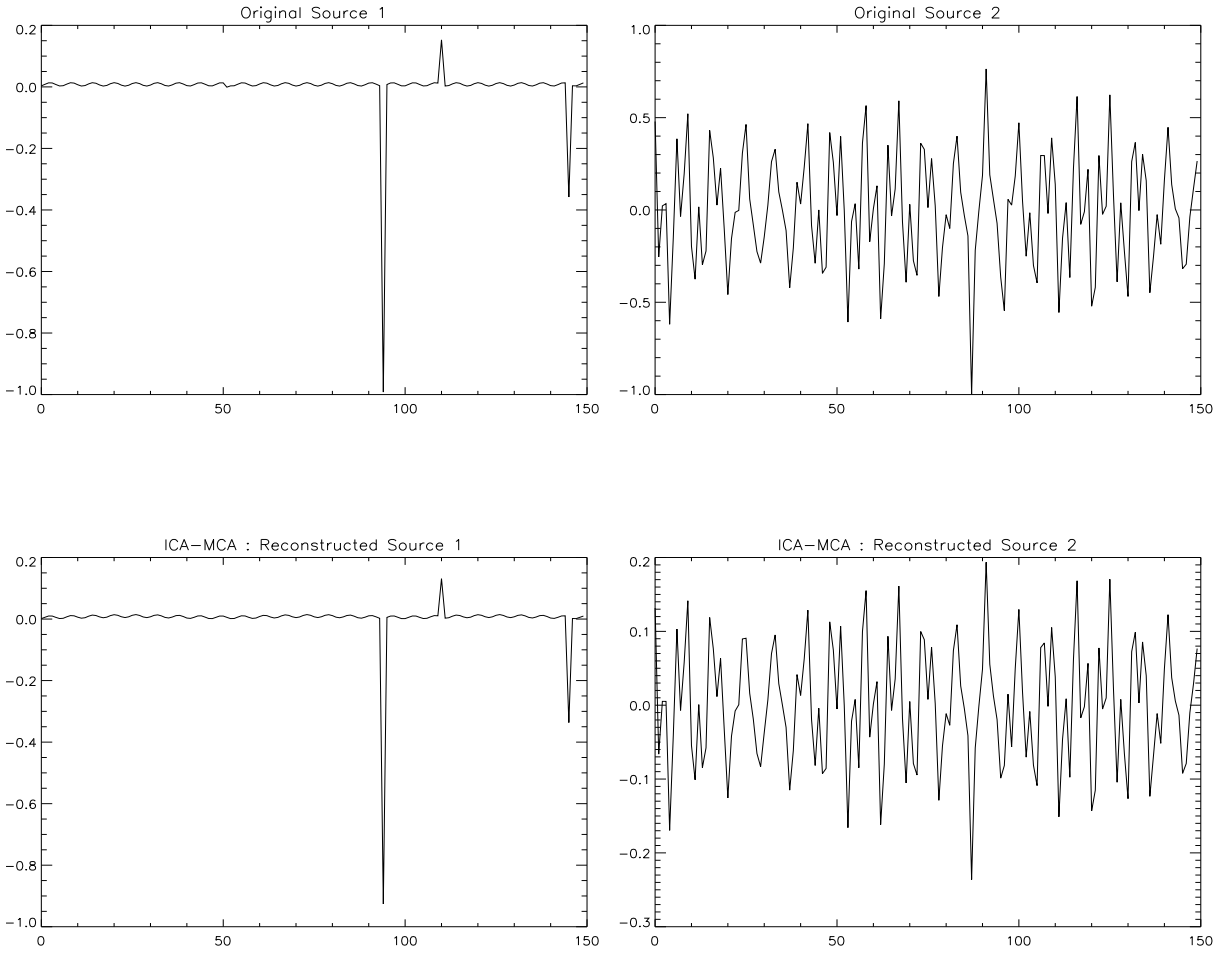



**Figure 6.** The observed data. Each channel is a linear combination of two sources plus a gaussian noise of standard deviation 0.03.



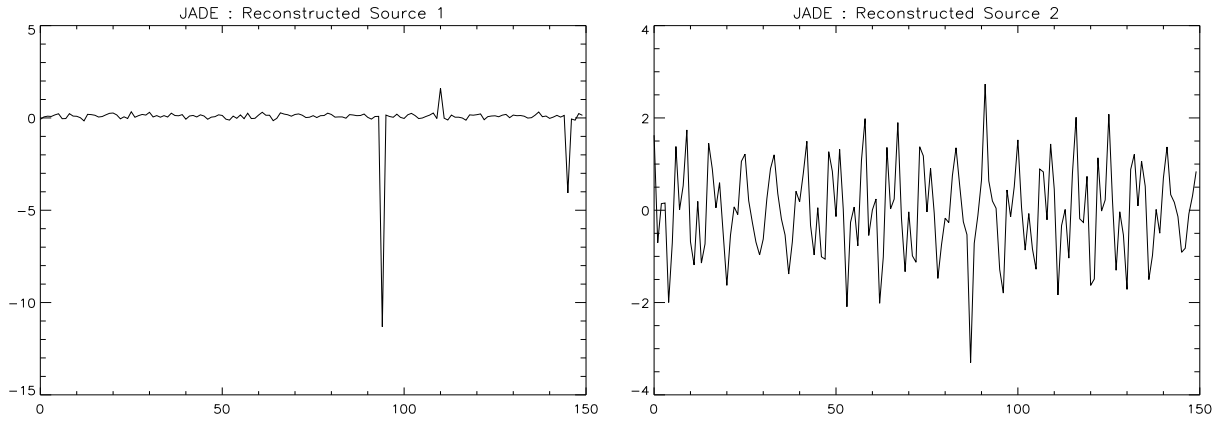



**Figure 7.** The top figures are the two observed data. Each channel is a linear combination of the two sources plus some Gaussian noise. At the bottom, the sources estimated using JADE. One can notice that the low frequency sinusoid of the first source is not well detected. In the middle, the figures show the sources estimated using ICA-MCA; the base line of the second source is now well estimated.

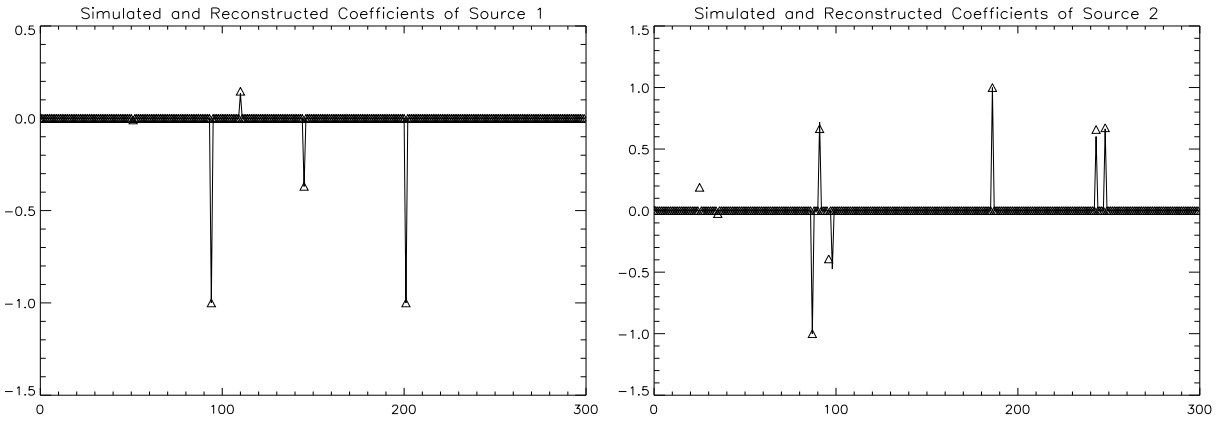



**Figure 8.** On the left, the simulated (△) and reconstructed (using ICA-MCA) coefficients of source 1. On the right, the simulated (△) and reconstructed (using ICA-MCA) coefficients of source 2. Each decomposition is achieved on the whole dictionnary : samples 0 to 149 correspond to the Dirac basis and samples 150 to 299 correspond to the DCT basis. Due to the scale indeterminacy discussed previously, only their relative amplitudes of the coefficients are significant.

7. M. Elad and A. Bruckstein, "A generalized uncertainty principle and sparse representation in pairs of bases," *IEEE Transactions on Information Theory* **48**, pp. 2558–2567, 2002.
8. R. Gribonval and M. Nielsen, "Some remarks on nonlinear approximation with schauder bases," *East J. on Approx.* **7**(2), pp. 267–285, 2001.
9. D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika* **81**, pp. 425–455, 1994.
10. A. Bruce, S. Sardy, and P. Tseng, "Block coordinate relaxation methods for nonparametric signal denoising," *Proceedings of the SPIE - The International Society for Optical Engineering* **3391**, pp. 75–86, 1998.
11. J.-L. Starck, F. Murtagh, and A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*, Cambridge University Press, 1998.
12. E. Candès and D. Donoho, "Ridgelets: the key to high dimensional intermittency?," *Philosophical Transactions of the Royal Society of London A* **357**, pp. 2495–2509, 1999.
13. M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE Trans. on Image Processing* **12**, pp. 882–889, 2003.
14. Y. Meyer, "Oscillating patterns in image processing and non linear evolution equations," *University Lecture Series Volume 22, AMS* , 2002.
15. L. Vese and S. Osher, "Modeling textures with total variation minimization and oscillating patterns in image processing," *Journal of Scientific Computing* **19**(1-3), pp. 553–572, 2003. in press.
16. J. Aujol, G. Aubert, L. Blanc-Feraud, and A. Chambolle, "Image decomposition: Application to textured images and sar images," Tech. Rep. ISRN I3S/RR-2003-01-FR, INRIA - Project ARIANA, Sophia Antipolis, 2003.
17. L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation noise removal algorithm," *Physica D* **60**, pp. 259–268, 1992.
18. J.-L. Starck, M. Elad, and D. Donoho, "Redundant multiscale transforms and their application for morphological component analysis," *Advances in Imaging and Electron Physics* **132**, 2004.
19. J.-L. Starck, M. Elad, and D. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Transactions on Image Processing* , 2004. in press.
20. M. Elad, J.-L. Starck, D. Donoho, and P. Querre, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *ACHA* , 2005. to appear.
21. C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and grey levels," *IEEE Trans. Image Processing* **10**, pp. 1200–1211, August 2001.
22. M. Bertalmio, A. Bertozzi, and G. Sapiro, "Navier-stokes,fluid dynamics, and image and video inpainting," in *in Proc.IEEE Computer Vision and Pattern Recognition (CVPR)*, 2001.
23. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," *Comput. Graph.(SIGGRAPH 2000)* , p. 417424, July 2000.
24. T. Chan and J. Shen, "Local inpainting models and tv inpainting," *SIAM J. Appl. Math.* **62**, pp. 1019–1043, July 2001.
25. J.-F. Cardoso, "The three easy routes to independent component analysis; contrasts and geometry," in *Proc. ICA 2001, San Diego*, 2001.
26. J.-F. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation* **11**, pp. 157–192, Jan. 1999.
27. A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley, New York, 2001. 481+xxii pages.
28. J. Delabrouille, J.-F. Cardoso, and G. Patanchon, "Multi–detector multi–component spectral matching and applications for CMB data analysis," *Monthly Notices of the Royal Astronomical Society* **346**, pp. 1089–1102, Dec. 2003. to appear, also available as http://arXiv.org/abs/astro-ph/0211504.
29. M. Zibulevsky and B. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," *Neural-Computation* **13**, pp. 863–882, April 2001.
30. R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *IEEE Transactions on Information Theory* **49**(12), pp. 3320–3325, 2003.