Morphological Diversity and Source Separation

J. Bobin, Y. Moudden, J.-L. Starck, and M. Elad

Abstract— This paper describes a new method for blind source separation, adapted to the case of sources having different morphologies. We show that such morphological diversity leads to a new and very efficient separation method, even in the presence of noise. The algorithm, coined MMCA (Multichannel Morphological Component Analysis), is an extension of the Morphological Component Analysis method (MCA). The latter takes advantage of the sparse representation of structured data in large overcomplete dictionaries to separate features in the data based on their morphology. MCA has been shown to be an efficient technique in such problems as separating an image into texture and piecewise smooth parts or for inpainting applications. The proposed extension, MMCA, extends the above for multichannel data, achieving a better source separation in those circumstances. Furthermore, the new algorithm can efficiently achieve good separation in a noisy context where standard ICA methods fail. The efficiency of the proposed scheme is confirmed in numerical experiments.

Index Terms— Blind source separation, sparse representations, morphological component analysis.

I. INTRODUCTION

A common assumption in signal or image processing is that measurements $\mathbf{X}$ made typically using an array of sensors, often consist of mixtures of contributions from various possibly independent underlying physical processes $\mathbf{S}$. The simplest mixture model is linear and instantaneous and takes the form:

$$\mathbf{X} = \mathbf{A}\,\mathbf{S} + \mathbf{N} \tag{1}$$

where $\mathbf{X}$ and $\mathbf{S}$ are random matrices of respective sizes $m \times t$ and $n \times t$ and $\mathbf{A}$ is an $m \times n$ matrix. Multiplying $\mathbf{S}$ by $\mathbf{A}$ linearly mixes the n sources into m observed processes. Thus, the rows of $\mathbf{S}$, s_k, are the sources, and the rows of $\mathbf{A}$, a_k, are the mixture weights. An $m \times t$ random matrix $\mathbf{N}$ is included to account for instrumental noise or model imperfections. The problem is to invert the mixing process so as to separate the data back into its constitutive elementary building blocks.

In the blind approach (where both the mixing matrix and the sources are unknown), and assuming minimal prior knowledge on mixing process, source separation is merely about devising quantitative measures of diversity or contrast. Classical Independent Component Analysis (ICA) methods assume that the mixed sources are statistically independent; these techniques (for example JADE, FastICA, Infomax) have proven to be successful in a wide range of applications (see [1]–[4], and references therein). Indeed, although statistical independence is a strong assumption, it is in many cases physically plausible.

An especially important case is when the mixed sources are highly sparse, meaning that each source is rarely active

J. Bobin★ (E-mail: jbobin@cea.fr), Y. Moudden (E-mail: ymoudden@cea.fr) and J-L. Starck are with the DAPNIA-SEDI-SAP, Service d'Astrophysique, CEA/Saclay, 91191 Gif sur Yvette, France. Phone:+33(0)169085273. Fax:+33(0)169086577.

J-L. Starck is also with Laboratoire APC, 11 place Marcelin Berthelot 75231 Paris Cedex 05. E-mail: jstarck@cea.fr.

M. Elad is with The Computer Science Department, The Technion - Israel Institute of Technology - Haifa 32000, Israel. E-mail:elad@cs.technion.ac.il.

and mostly (nearly) zero. The independence assumption in such case implies that the probability for two sources to be significant simultaneously is extremely low, so that the sources may be treated as having nearly disjoint supports. This is exploited for instance in sparse component analysis [5]. Indeed, it has been already shown in [6] that first moving the data into a representation in which the sources are assumed to be sparse greatly enhances the quality of the separation. Possible representation dictionaries include the Fourier and related bases, wavelet bases, and more. Working with combinations of several bases or with very redundant dictionaries such as the undecimated wavelet frames or the more recent ridgelets and curvelets [7] could lead to even more efficient representations. However, finding the smallest subset of elements (that linearly combine to reproduce a given signal or image) is a hard combinatorial problem. Nevertheless, several pursuit algorithms have been proposed that can help build very sparse decompositions [8], [9]. In fact, a number of recent results prove that these algorithms will recover the unique optimal decomposition provided that this solution is sparse enough and the dictionary is sufficiently incoherent [10], [11].

In another context, the Morphological Component Analysis (MCA) described in [12] uses the idea of sparse representation for the separation of sources from a single mixture. MCA constructs a sparse representation of a signal or an image considering that it is a combination of features which are sparsely represented by different dictionaries. For instance, images commonly combine contours and textures: the former are well accounted for using curvelets, while the latter may be well represented using local cosine functions. In searching a sparse decomposition of a signal or image y, it is assumed that y is a sum of n components, s_k, where each can be described as $s_k = \mathbf{\Phi}_k \alpha_k$ with an over-complete dictionary $\mathbf{\Phi}_k$ and a sparse representation α_k. It is further assumed that for any given component the sparsest decomposition over the proper dictionary yields a highly sparse description, while its decomposition over the other dictionaries, $\mathbf{\Phi}_{k' \neq k}$, is highly non sparse. Thus, the different $\mathbf{\Phi}_k$ can be seen as discriminating between the different components of the initial signal. Ideally, the α_k are the solutions of :

$$\min_{\{\alpha_1, \ldots, \alpha_n\}} \sum_{k=1}^{n} \|\alpha_k\|_0 \quad \text{subject to} \quad y = \sum_{k=1}^{n} \mathbf{\Phi}_k \alpha_k. \tag{2}$$

However, as the ℓ^0 norm is non-convex, optimizing the above criterion is combinatorial by nature. Substituting the ℓ^0-norm by an ℓ^1, as motivated by recent equivalence results [10], and relaxing the equality constraint, the MCA algorithm seeks a solution to

$$\min_{y_1, \ldots, y_n} \lambda \sum_{k=1}^{n} \|\alpha_k\|_1 + \left\| y - \sum_{k=1}^{n} s_k \right\|_2^2 \quad \text{with} \quad s_k = \mathbf{\Phi}_k \alpha_k. \tag{3}$$

A detailed description of MCA is given in [12] along with results of experiments in contour/texture separation in images and inpainting. Note that there is no mixing matrix to be estimated in the MCA model and the mixture weights are absorbed by the source signals s_k.

The purpose of this contribution is to extend the MCA to the

case of multi-channel data, as described in the next section. In handling several mixtures together, the mixing matrix becomes an unknown as well, which adds some complexity to the overall problem. On the other hand, having more than one mixture is expected to help the separation, leading to better performance compared to regular MCA. Section III illustrates the performance of MMCA, and demonstrates its superiority over both MCA and several ICA techniques. We should note that our method could also be considered as an extension of the algorithm described in [6], with two major differences: (i) while [6] uses a a single transform to sparsify the data, our technique assumes the use of different dictionaries for different sources; (ii) the numerical scheme that we lead to in the construction of the algorithm is entirely different. Interestingly, a similar philosophy has been employed by [13] for audiophonic signals. Their method assumes that an audio signal is mainly made of a 'tonal' part (sparse in a discrete cosine dictionary), a transient part (well sparsified by a wavelet transform) and a residual. However their decomposition algorithm is not based on an iterative scheme, which is a major difference with MMCA. Indeed, experiments show that such an iterative process is needed when the considered transforms are far from being incoherent (for instance DCT and curvelet transform).

II. MULTICHANNEL MCA

We consider the mixing model (1) and make the additional assumption that each source s_k is well (*i.e.* sparsely) represented by a specific and different dictionary $\mathbf{\Phi}_k$. Assigning a Laplacian prior with precision λ_k to the decomposition coefficients of the kth source s_k in dictionary $\mathbf{\Phi}_k$ is a practical way to implement this property. Here, s_k denotes the $1 \times t$ array of the kth source samples. Classically, we assume zero-mean Gaussian white noise. This leads to the following joint estimator of the source processes $\mathbf{S} = \left[s_1^T, \ldots, s_n^T\right]^T$ and the mixing matrix $\mathbf{A}$:

$$\{\hat{\mathbf{S}}, \hat{\mathbf{A}}\} = \text{Arg} \min_{\mathbf{S}, \mathbf{A}} \|\mathbf{X} - \mathbf{AS}\|_F^2 + \sum_k \lambda_k \|s_k \mathbf{T}_k\|_1, \quad (4)$$

where $\|\mathbf{M}\|_F^2 = \text{trace}\left(\mathbf{M}^T \mathbf{M}\right)$ is the Frobenius norm. In the above formulation we define $\mathbf{T}_k = \mathbf{\Phi}_k^+$, implying that the transform is applied in an analysis mode of operation, very much like in the MCA [12]. Unfortunately, this minimization problem suffers from a lack of scale invariance of the objective function: scaling the mixing matrix by $\mathbf{A} \leftarrow \rho \mathbf{A}$, and an inverse scaling of the source matrix, $\mathbf{S} \leftarrow \frac{1}{\rho}\mathbf{S}$, leaves the quadratic measure of fit unchanged while deeply altering the sparsity term. This problem can be alleviated by forcing the mixing matrix to have normalized columns a^k, implying that each of the source signals is scaled by a scalar. Practically, this can be achieved by normalizing these columns at each iteration ($a^{k^+} \leftarrow a^{k^-}/\|a^{k^-}\|_2$), and propagating the scale factor to the corresponding source by $s_k{}^+ \leftarrow \|a^{k^-}\|_2 s_k{}^-$. We propose solving (4) by breaking $\mathbf{AS}$ into n rank-1 terms, $\mathbf{AS} = \sum_{k=1}^n a^k s_k$, and updating one at a time. Define the kth multichannel residual $\mathbf{X}_k = \mathbf{X} - \sum_{k' \neq k} a^{k'} s_{k'}$ as corresponding to the part of the data unexplained by the other couples $\{a^{k'}, s_{k'}\}_{k' \neq k}$. Then, minimizing the objective

function with respect to s_k assuming a^k is fixed as well as all $s_{k' \neq k}$ and $a^{k' \neq k}$ leads to:

$$s_k = \frac{1}{\|a^k\|_2^2}\left(a^{k^T}\mathbf{X}_k - \frac{\lambda_k}{2}\text{Sign}(s_k \mathbf{T}_k)\mathbf{T}_k^T.\right) \quad (5)$$

This is a closed-form solution, known as soft-thresholding, known to be exact for the case of unitary matrices $\mathbf{T}_k$.

As $\mathbf{T}_k$ becomes a redundant transform, we keep this interpretation as an approximate solution, and update the source signal s_k by soft-thresholding the coefficients of the decomposition of a *coarse version* $(1/\|a^k\|_2^2)a^{k^T}\mathbf{X}_k\mathbf{T}_k$ with a scalar threshold $\lambda_k/(2\|a^k\|_2^2)$ (see [14] for more details on the justification of this step). Then, considering a fixed s_k, the update on a^k follows from a simple least squares linear regression. The MMCA algorithm is given below:

---
1. Set number of iterations $L_{\max}$ & thresholds $\forall k, \delta_k = L_{\max} \cdot \lambda_k/2$
2. While $\delta_k > \lambda_k/2$,
   For $k = 1, \ldots, n_s$:
   - Renormalize a^k, s_k and δ_k
   - Update s_k assuming all $s_{k' \neq k}$ and $a^{k'}$ are fixed:
     – Compute the residual $\mathbf{X}_k = \mathbf{X} - \sum_{k' \neq k} a^{k'} s_{k'}$
     – Project $\mathbf{X}_k$: $\tilde{s}_k = \frac{1}{\|a^k\|_2^2}a^{k^T}\mathbf{X}_k$
     – Compute $\alpha_k = \tilde{s}_k \mathbf{T}_k$
     – Soft threshold α_k with threshold δ_k, yielding $\hat{\alpha}_k$
     – Reconstruct s_k by $s_k = \hat{\alpha}_k \mathbf{T}_k^T$
   - Update a^k assuming all $s_{k'}$ and $a^{k' \neq k}$ are fixed:
     – $a^k = \frac{1}{\|s_k\|_2^2}\mathbf{X}_k s_k{}^T$
   Lower the thresholds: $\delta_k = \delta_k - \lambda_k/2$.

At each iteration, *coarse* (*i.e. smooth*) versions of the sources are computed. The mixing matrix is then estimated from sources that contain the most significant parts of the original sources. The overall optimization proceeds by alternately refining both the sources and the mixing matrix. The use of a progressive thresholding scheme with a set of thresholds δ_k decreasing slowly towards $\lambda_k/2$ enforces a certain robustness to noise. Indeed, both alternate projections and iterative thresholding define a non trivial path for the variables to estimate (sources and mixing matrix) during the optimization. This optimization scheme leads to a good estimation as underlined in [12]. MMCA benefits from the potential of overcomplete dictionaries for sparse representation. In comparison with the algorithm in [6], which uses a single sparsifying transform and a quadratic programming technique, our method considers more than just one transform and a shrinkage-based optimization. In the case where we have only one channel and the mixing matrix is known and equal to $(1 \cdots 1)$ then we can see that MMCA is equivalent to MCA. The next section will illustrate the efficiency of the MMCA algorithm when the sources to be separated have different morphologies.

III. RESULTS

A. Experiment 1: One-dimensional toy example

We start by illustrating the performance of MMCA with the simple BSS experiment on one-dimensional data. The two source signals at the top-left of Figure 1 were linearly mixed to form the three synthetic observations shown at the top-right. A Gaussian noise with $\sigma = 0.05$ was also added to

the mixtures (note that each channel has a unit variance). The two sources are morphologically different: one consists of four bumps and the other is a plain sine-wave. Source separation was conducted using the above MMCA algorithm, using the Fourier and the trivial basis as the representing dictionaries. For the sake of comparison, a publicly available implementation of the JADE algorithm was also tested[1]. As can be seen, MMCA is clearly able to efficiently separate the original source signals. Note that denoising is an intrinsic part of the algorithm.
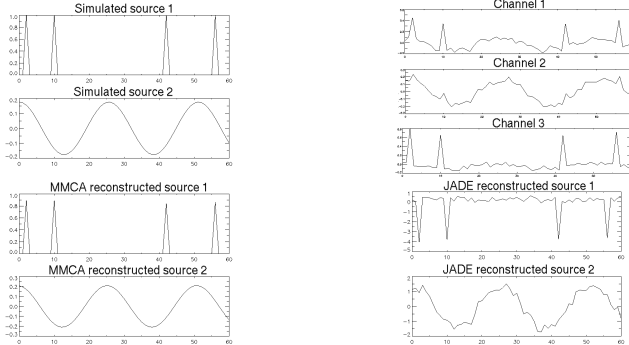
Fig. 1. Experiment 1: **Top left:** The two initial source signals. **Top right:** Three noisy *observed* mixtures. **Bottom left:** The two source signals reconstructed using MMCA. **Bottom right:** The two source signals reconstructed with Jade.

B. Experiment 2: Blind separation of images

We now turn to use MMCA to separate efficiently two-dimensional data. In Figure 2, the two left pictures are the sources. The first source image is composed of three curves which are well represented by a curvelet transform. We use the global discrete cosine transform (DCT) to represent the second source image. Although the resulting representation may not be extremely sparse, what is significant here is that contrastingly the representation of the first component using the global DCT is not sparse. The mixtures are shown in the second image pair. A Gaussian noise has been added to these mixtures, using different noise variances for the different channels. Finally the two images in the last column show the MMCA source estimates. Visually the MMCA performs well.

We compare the MMCA with two standard source separation techniques: JADE and FastICA [1]. As the original JADE algorithm has not been devised to take into account additive noise, we apply denoising on its outputs (using a standard wavelet denoising technique assuming that the noise variances are known). Note that we could denoise the data before separation; however the non-linear wavelet denoising erases the coherence between the channels, so that an ICA-based method would fail to separate the sources from the denoised data. We also compare MMCA with a more recent method based on sparse representations which is described in [15]. We also estimate the mixing matrix using the Relative Newton Method after a 2D-wavelet transform of the mixtures. The graphs in Figure 3 show the correlation between the original sources and their estimates as the data noise variance increases. One can note that both JADE and FastICA have similar performance. As the data noise variance increases, MMCA

[1]Taken from *http://www.tsi.enst.fr/ cardoso/guidesepsou.html*.

clearly achieves better source estimation, and shows clear robustness compared to non-denoised ICA-based methods and to the Relative Newton Method. We also observed that the Relative Newton Method [15] seems rather unstable as the noise variance increases. MMCA provides a similar behavior compared to denoised versions of the classical ICA-based algorithms.

As the noise variance increases, the mixing matrices estimated using ICA-based methods are biased and thus these methods fail to correctly estimate the sources. Moreover, denoising after the separation process softens the separation error. Hence, the denoised versions of the JADE and FastICA seem to perform as well as MMCA. As a consequence, a more efficient criterion is needed. A natural way of assessing the separation quality is to compare the estimated and original mixing matrices. Quantitative results are shown in Figure 4, where the mixing matrix estimation error is defined as $\rho_{\mathbf{A}} = ||\mathbf{I} - \mathbf{\Lambda}^{-1}\tilde{\mathbf{A}}^{-1}\mathbf{A}||_1$ (vector norm). $\mathbf{A}$ is the true mixing matrix, $\tilde{\mathbf{A}}$ is the estimated one and $\mathbf{\Lambda}$ is a matrix which restores the right scaling and permutation on the estimated matrix. If $\tilde{\mathbf{A}} = \mathbf{\Lambda}\mathbf{A}$ (*i.e* $\tilde{\mathbf{A}}$ is equal to $\mathbf{A}$ up to scaling and permutation) then $\rho_{\mathbf{A}} = 0$; thus $\rho_{\mathbf{A}}$ measures a deviation from the true mixture. Contrasting with standard ICA methods, MMCA iteratively estimates the mixing matrix $\mathbf{A}$ from coarse (*i.e.* smooth) versions of the sources and thus is not penalized by the presence of noise. As a consequence, MMCA is clearly more robust to noise than standard ICA methods even in the case of very noisy mixtures. Indeed it can be noticed in Figure 3 and 4 that when the noise variance increases, standard ICA-based methods fail whereas MMCA still performs well. MMCA also performs better than a sparsity-based algorithm described in [15].
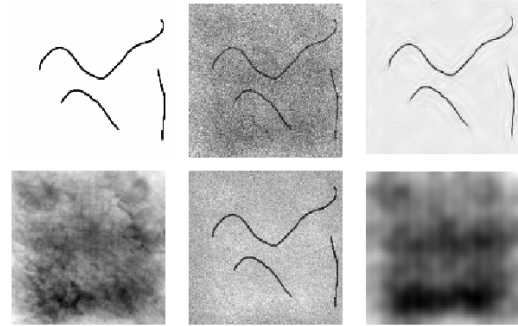

Fig. 2. Experiment 2 (using curvelet and DCT): **First column:** The original sources of variance 1. **Second column:** their mixtures (a Gaussian noise is added : $\sigma = 0.4$ and 0.6 for channels 1 and 2 respectively. The mixtures are such that $x_1 = 0.5s_1 - 0.5s_2$ and $x_2 = 0.3s_1 + 0.7s_2$).**Third column:** sources estimated by MMCA.
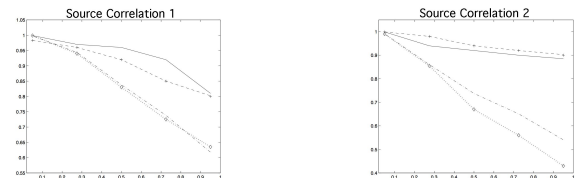

Fig. 3. Experiment 2: Correlation between the true source signals and the sources estimated by JADE (*dotted line*), denoised JADE (*dashed line*), FastICA ($\diamond$), denoised FastICA ($+$), the Relative Newton Method (*dashdot*) and MMCA (*solid*), as a function of the noise power σ.

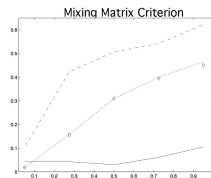Fig. 4. Experiment 2: Mixing matrix error (defined via ρ_A) for JADE (*dotted line*), FastICA ($\diamond$), the Relative Newton Method (*dashdot*) and MMCA (*solid*), as a function of the noise power σ.

C. Experiment 3: MMCA versus MCA

Morphological Component Analysis [12] has been devised to extract both texture and cartoon components from a single image. We describe here an experiment where we use MMCA for a similar purpose in order to compare the two methods. Note that MCA is applied when only one mixture ($m = 1$) is provided. Let us point out that the main difference between these methods is the estimation of the mixing matrix in MMCA which is not needed in MCA. Figure 5 features two original pictures: the first one is mainly a cartoon well sparsified by a curvelet transform; the other is a texture represented well by global 2D-DCT. Two noisy mixtures are shown in the second column. We applied MCA to the sum of the two original sources, and MMCA to a random number of mixtures (between 2 and 10 channels). The last column of Figure 5 features the two sources estimated by MMCA based on 10 mixtures. Quantitatively, Figure 6 shows the correlation between the original sources and those estimated using MMCA as the number of mixtures increases. Clearly, the amount of information provided by the multichannel data improves source estimation, as expected.
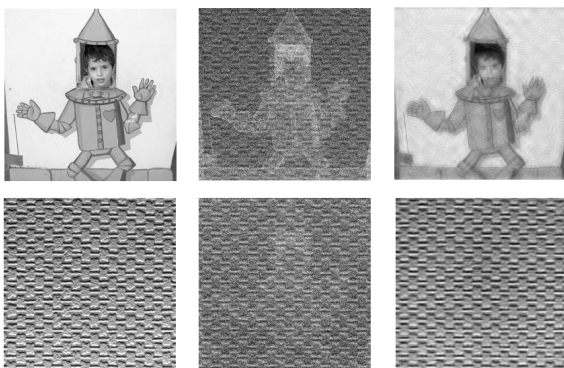

Fig. 5. Experiment 3 (using curvelet and DCT): **First column:** The original sources. They have been normalized to unit variance. **Second column:** mixtures of the initial sources. A Gaussian noise of variance $\sigma = 0.3$ was added to each channel. **Third column:** sources estimated by MMCA from 10 mixtures.

IV. CONCLUSION

The MCA algorithm provides a powerful and fast signal decomposition method, based on sparse and redundant representations over separate dictionaries. The MMCA algorithm described in this paper extends MCA to the multichannel case. For blind source separation, this extension is shown to perform well provided the original sources are morphologically different, meaning that the sources are sparsely represented in different bases. We also demonstrated that MMCA performs better than standard ICA-based source separation in a noisy context. We are currently working on improvements and generalizations of MMCA where each source can be modelled as a linear combination of morphologically different components.

REFERENCES

[1] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: John Wiley, 2001, 481+xxii pages. [Online]. Available: http://www.cis.hut.fi/projects/ica/book/

[2] A. Belouchrani, K. A. Meraim, J.-F. Cardoso, and E. Moulines, "A blind source separation technique based on second order statistics," *IEEE Trans. on Signal Processing*, vol. 45, no. 2, pp. 434–444, 1997.

[3] D.-T. Pham and J.-F. Cardoso, "Blind separation of instantaneous mixtures of non stationary sources," *IEEE Trans. on Sig. Proc.*, vol. 49, no. 9, pp. 1837–1848, Sept. 2001.

[4] 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*, vol. 346, no. 4, pp. 1089–1102, Dec. 2003, to appear, also available as http://arXiv.org/abs/astro-ph/0211504.

[5] P. G. Georgiev, F. Theis, and A. Cichocki, "Sparse component analysis and blind source separation of underdetermined mixtures," *IEEE Transactions on Neural Networks*, vol. 16, no. 4, pp. 992–996, 2005.

[6] M. Zibulevsky and B. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," *Neural-Computation*, vol. 13, no. 4, pp. 863–882, April 2001.

[7] J.-L. Starck, E. Candès, and D. Donoho, "The curvelet transform for image denoising," *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 131–141, 2002.

[8] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[9] S. Chen, D. Donoho, and M. Saunder, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[10] D. L. Donoho and M. Elad, "Maximal sparsity representation via l_1 minimization," *the Proc. Nat. Aca. Sci.*, vol. 100, pp. 2197–2202, 2003.

[11] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3320–3325, 2003.

[12] J.-L. Starck, M. Elad, and D. Donoho, "Redundant multiscale transforms and their application for morphological component analysis," *Advances in Imaging and Electron Physics*, vol. 132, 2004.

[13] L. Daudet and B. Torresani., "Hybrid representations for audiophonic signal encoding," *Signal Processing, Special issue on Image and Video Coding Beyond Standards*, vol. 82, pp. 1595–1617, 2002.

[14] M. Elad, "Why simple shrinkage is still relevant for redundant representations?" submitted to the IEEE Trans. On Information Theory on January 2005.

[15] M. Zibulevski, "Blind source separation with relative newton method," *Proccedings ICA2003*, pp. 897–902, 2003.
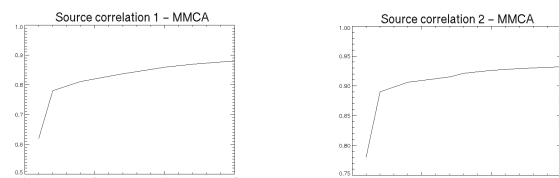
Fig. 6. Experiment 3: Correlation between the true sources and the MMCA estimates as the number of mixtures increases, **Left:** cartoon component - **Right:** texture component. Note that the results for one mixture correspond to MCA.