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

IMAGE SEPARATION USING ITERATED POSTERIOR POINT ESTIMATION

Koray Kayabol[1], *Ercan E. Kuruoglu*[2], *and Bulent Sankur*[3]

[1]Electrical and Electronics Engineering Dept., Istanbul University,
Avcilar, 34320, Istanbul, Turkey
email: kayabol@istanbul.edu.tr
[2]Instituto di Scienza e Tecnologie dell'Informazione, CNR,
via G. Moruzzi 1, 56124, Pisa, Italy
email: ercan.kuruoglu@isti.cnr.it
[3]Electrical and Electronics Engineering Dept., Bogazici University,
Bebek, 80815, Istanbul, Turkey
email: bulent.sankur@boun.edu.tr

ABSTRACT

In this study, we introduce a new method for image separation problem using Monte Carlo (MC) integration method. The proposed method stands between Gibbs sampling and deterministic optimization based Iterated Conditional Mode (ICM) methods. In this sense, it incorporates the better of the two paradigms, in that it is 2 to 3 times faster than Gibbs sampling and shows better performance compared to ICM. The novelty of the method consists in the use of the conditional expectation of some robust error function as a cost function for pixels. The point estimate, that is the minimum of the robust error function, is found iteratively using a gradient descent algorithm. The stochastic gradient itself is computed using importance sampling since the posterior is not integrable analytically. Furthermore at each iteration, only the point estimates are saved, in contrast to sequential MC methods which save all of the particles (samples) of a single variable and evaluate them sequentially. The mixing matrix is estimated by the the Mean Square Error (MSE) algorithm, and the source images are modelled via Markov Random Fields (MRF).

1. INTRODUCTION

The separation of images from their mixtures given a number of observations can be interpreted as an inverse problem. The literature has witnessed several solutions to image separation based on the estimation of a de-mixing matrix using various contrast functions. Recently this problem has been addressed within the Bayesian paradigm. Notably Kuruoglu et al. [1] have used the Bayesian technique to image separation problem. They modelled the prior of the pixel values with a MRF. Tonazzini et al. [2] applied mean field approximation to MRF and used EM algorithm to estimate the sources and the mixing matrix parameters. [3] applied a variational approximation to image priors modelled as MRF, with the purpose of converting a non-convex problem into a convex one.

These methods achieve source separation by finding the MAP estimates of all the variables, that is, mixing coefficients, pixels of all components and additive noise variance. The MAP estimate is, however, convenient for unimodal posterior densities because otherwise it becomes very tedious to find the global maximum point among local maxima. A way to overcome this impasse is Markov chain Monte Carlo (MCMC) methods [4]. For example Gibbs sampling, which is one of the MCMC methods, reaches a solution by drawing random samples from posterior densities instead of finding local maxima of the corresponding density. By drawing samples from the estimated posterior density, the solution space is better explored especially after the initial transient estimates (the burn-in period).

The proposed MC algorithm consists in sampling the full density rather than generating single samples to stalk local maxima. Our approach is essentially applying particle method to estimate the a-posteriori distribution on a pixel by pixel basis. In other words, we do not estimate pixels directly, but we first estimate their pdf. The advantage is that, once the a posterior pdf's are available, one can attain diverse estimators that range from MSE to error entropy minimizer. We exploited a cost function which is formed by taking the expectation of a robust error function with respect to posterior density. The general case of this cost function can be found in [5]. In [5], a pdf is assigned to the error and an information theoretic cost function is used. A closed form solution using our cost function is not readily available, so we used a gradient descent type iterative method. But the gradient of the cost function contains a stochastic integral. We overcome the calculation of the integral using a Monte Carlo integration technique. In [5], the Parzen density estimation was used for calculating integral. The proposed method can be seen as a generalized version of the Least Mean Square (LMS) algorithm.

In particle filtering methods, the purpose is to calculate the integral by Monte Carlo methods for finding the point estimates of variables. The basic deterministic method for calculating integrals numerically is the finite difference approximation. In this method, the estimate is obtained by sampling the pdf of the variable uniformly, provided there is a sufficient number of samples. If the number of samples is not sufficient, the integral can still be accurately calculated by drawing samples from the highly probable regions, the so-called importance sampling (IS) scheme. This leads naturally to the computation of the integrals by using random samples. If generating samples from posterior density turns out to be difficult, the samples are produced from a proposal or importance density. For example in MRF-modelled images, it may not be easy to draw samples from the Gibbsian form of pixel densities.

The Sequential MC sampler (SMC) [6], where the signal is assumed to form a Markov process, is valid for 1D causal signals, and its direct application to 2D signals is not straightforward. If only the causal neighborhood of pixels is considered to condition the pixel pdf's, than half of the neighborhood information is sacrificed. Costagli et al. [7] suggest a partial solution to this problem by fusing multiple 1D particle filters. To recuperate the loss of information due to their 1D stratagem they scan the image in several different directions and then fuse these estimates.

In another 2D particle filter realization applied to image restoration, Nittono et al. [8] used the prior density of MRF-modelled image as the importance function. Zhai et al. [9] used particle representation of the pdf of the pixel intensitiesand saved particles of only one pixel at a time. They evaluate these particles using 1D importance function whose parameters are obtained by a Kalman filter. Our method also uses the MRF model for images, but it differs from previous studies in [8, 9] in that we use a robust point estimator denoted as iterated posterior point estimator. We use Monte Carlo method to obtain the point estimates but do not save the particles of any one pixel in memory, but only its point estimate. Our second contribution is that we apply the proposed 2D particle method to image separation problem.

The rest of the paper is laid out as follows. In Section 2, the problem formulation will be presented in the Bayesian framework. In Section 3, the details of the iterated posterior point estimation algorithm will be revealed. The simulation results are given in Section 4.

2. PROBLEM FORMULATION

In this study, we use linear mixing model with additive Gaussian noise

$$\begin{bmatrix} \mathbf{y}_1^T \\ \mathbf{y}_2^T \\ \vdots \\ \mathbf{y}_K^T \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{s}_1^T \\ \mathbf{s}_2^T \\ \vdots \\ \mathbf{s}_L^T \end{bmatrix} + \mathbf{V} \tag{1}$$

where $\mathbf{y}_{1:K}$ and $\mathbf{s}_{1:L}$ are vector representations of observation and source images, respectively. If the size of images are denoted as $N = N_1 \times N_2$, the length of the vector representation of images becomes $N \times 1$. $\mathbf{V}$ is a $K \times N$ zero mean white noise matrix and $\mathbf{A}$ is $K \times L$ mixing matrix. The sources are assumed to be independent. So the joint probability density is factorized as $p(\mathbf{s}_{1:L}) = \prod_{l=1}^{L} p(\mathbf{s}_l)$.

The joint posterior density of unknowns is expressed as $p(\mathbf{s}_{1:L}, \mathbf{A}|\mathbf{y}_{1:K})$. The sources and the mixing matrix are estimated by maximizing the a posteriori probability, that is, by finding the MAP solution of the problem. But it is not straightforward to find a joint estimate. The ICM (Iterated Conditional Mode) algorithm has been used to reach joint solution by separating the problem into consecutive maximization steps for each variable. In our approach, we also separate the problem into consecutive steps but do not attempt to obtain the MAP estimate directly. Instead, we define a mean error function and find its solution iteratively using MC methods. The details will be given in the next section.

We need the posterior density of each variable for estimation. According to Bayes rule, the posterior of nth pixel of the lth source is written as

$$p(s_{l,n}|\mathbf{y}_{1:K}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) \propto p(\mathbf{y}_{1:K}|\mathbf{s}_{1:L}, \mathbf{A})p(s_{l,n}|\mathbf{s}_{1:L-(l,n)}) \tag{2}$$

where the notation $\mathbf{s}_{1:L-(l,n)}$ represents all of the source images pixels except the nth pixel of the lth source and $n = 1,\dots,N$. Since the image is modelled as MRF, the prior density of pixel n, $p(s_{l,n}|\mathbf{s}_{1:L-(l,n)}) = p(s_{l,n}|s_{\partial l,n})$, is in the Gibbs form where ∂n represents the first order spatial neighbors of the pixel. The Gibbs distribution is given as

$$p(s_{l,n}|s_{\partial l,n}) = \frac{1}{Z(\beta)} e^{-\frac{1}{2}\sum_{\partial l,n} \beta\rho(s_{l,n}-s_{\partial l,n})} \tag{3}$$

where the clique potential function $\rho(.)$ is chosen as the logarithm of the Cauchy density

$$\rho(s_n - s_m) = \ln\left[1 + \frac{(s_n - s_m)^2}{\delta}\right]. \tag{4}$$

where $m \in \partial n$.

The likelihood factor in (2) is Gaussian because of the noise model, and therefore the posterior is formed by multiplying a Gaussian and a Cauchy density. The MAP solution using this posterior expression can be found by optimization, for example, via some descent algorithm. However the convergence is not guaranteed since the clique potential is not convex. Gibbs sampling is a more proper approach in this case, though the computational complexity is significantly higher. Instead of applying a descent algorithm on (2) or reverting to Gibbs sampling, we propose an alternate robust approach, that of sampling the entire posterior using particle methods, as detailed in the next section.

The posterior of an element of the mixing matrix is given as

$$p(a_{k,l}|\mathbf{A}_{-(k,l)}, \mathbf{y}_{1:K}, \mathbf{s}_{1:L}) \propto p(\mathbf{y}_{1:K}|\mathbf{s}_{1:L}, a_{k,l}, \mathbf{A}_{-(k,l)}) \tag{5}$$

where $\mathbf{A}_{-(k,l)}$ is the part of $\mathbf{A}$ which does not contain the element (k,l). The uniform prior is chosen for $\mathbf{A}$.

3. ITERATED POSTERIOR POINT ESTIMATION

The particle methods represent the posterior pdf of a random variable by drawing some particles and assigning a weight to each of them. In sequential MC, this procedure is performed by evaluating only a single pdf sequentially. In this case only one sampled pdf is kept in the memory. For a 1D sequence this is all that is needed. But in the 2D case, saving the pdfs of the pixels already visited needs more memory and furthermore, producing new particles based on the particles of neighbors is not straightforward. In our approach, we save only point estimates of the pixels, and to this effect we use importance sampling.

We choose the minimum of the expected value of a robust error function, given in (6), as our point estimate. If an analytical form is not available, the minimum of the robust error function can be reached iteratively. We use at each iteration importance sampling on the posterior of the variable. The point estimation detail given for a single pixel of a source in (8) is applicable to all other pixels and the mixing matrix coefficients. The expectation of the robust error function of a pixel is given as:

$$\varepsilon = \int g(s_{l,n} - \hat{s}_{l,n}) p(s_{l,n}|\mathbf{y}_{1:K}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) ds_{l,n} \tag{6}$$

For times $t = 1 : T$
  For all source images, $l = 1 : L$
    For all pixels, $n = 1 : N$
      • For $i = 1 : I$, sample $s_{l,n}^{(i)} \sim \mathscr{U}(s_{l,n} \mid \hat{s}_{l,n} - \theta, \hat{s}_{l,n} + \theta)$
      • Evaluate the importance weights $w_{l,n}^{(i)}$ using 11
      • Normalize the importance weights $\bar{w}_{l,n}^{(i)}$ using 12
      • Calculate the new point estimate $\hat{s}'_{l,n}$ using 8 and 10
  For all elements of the mixing matrix, $(k,l) = (1,1) : (K,L)$
    • For $j = 1 : J$, sample $a_{k,l}^{(j)} \sim p(\mathbf{y}_{1:K} \mid \mathbf{s}_{1:L}, a_{k,l}, \mathbf{A}_{-(k,l)})$
    • Calculate the MSE estimate $\hat{a}'_{k,l}$ using 14

Table 1: The iterated posterior point estimation algorithm.

where the $g(.)$ is a robust function for error. If the $g(.) = |.|^2$ which is the mean squared error case, the solution can be found analytically as

$$\begin{aligned} \hat{s}_{l,n} &= E[s_{l,n} \mid \mathbf{y}_{1:K}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}] \\ &= \int s_{l,n} p(s_{l,n} \mid \mathbf{y}_{1:K}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) ds_{l,n} \end{aligned} \tag{7}$$

This is the MSE estimate of s_n. But in a more general case as in (6), we can find the solution using iterative techniques. If the steepest descent method is used, we can write the one step iteration of s_n as

$$\hat{s}'_{l,n} = \hat{s}_{l,n} - \mu \int g'(s_{l,n} - \hat{s}_{l,n}) p(s_{l,n} \mid \mathbf{y}_{1:K}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) ds_{l,n} \tag{8}$$

where $\hat{s}'_{l,n}$ is the new estimation, $g'(.)$ is the first derivative of $g(.)$, and the μ is the step size of the descent method. We choose the robust function as $g(.) = |.|^p$ where $1 < p \leq 2$.The posterior density expression in the integrand is formed as

$$\begin{aligned} p(s_{l,n} \mid \mathbf{y}_{1:K}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) \propto \quad & p(\mathbf{y}_{1:K} \mid s_{l,n}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) \times \\ & p(s_{l,n} \mid s_{\partial l,n}) \end{aligned} \tag{9}$$

If we consider the SMC sampling, the likelihood term $p(\mathbf{y}_{1:K} \mid s_{l,n}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A})$, corresponds to observation probability and the prior of $s_{l,n}$, $p(s_{l,n} \mid s_{\partial l,n})$, corresponds to transition probability. In this case, the prior density define the spatial transition probability of a 2D MRF.

To calculate the integral in (8), we use Monte Carlo integration. The direct sampling from posterior density is not straightforward, so we can use MCMC techniques for this. For avoiding the difficulty of MCMC, we use the importance sampling method. For this purpose, we choose a importance density, $q(s_{l,n} \mid \mathbf{s}_{1:L-(l,n)}, \hat{s}_{l,n}, \mathbf{y}_{1:K}, \mathbf{A})$, and draw the random samples from this density. The approximate integral is written as

$$\int g'(s_{l,n} - \hat{s}_{l,n}) \frac{p(s_{l,n} \mid .)}{q(s_{l,n} \mid .)} q(s_{l,n} \mid .) ds_{l,n} \approx$$

$$\sum_{i=1}^{I} \bar{w}_{l,n}^{(i)} g'(s_{l,n}^{(i)} - \hat{s}_{l,n}) \tag{10}$$

where I is the number of samples and

$$w_{l,n}^{(i)} = \frac{p(\mathbf{y}_{1:K} \mid s_{l,n}^{(i)}, \mathbf{s}_{1:L-(l,n)}, \mathbf{A}) p(s_{l,n}^{(i)} \mid s_{\partial l,n})}{q(s_{l,n} \mid \mathbf{s}_{1:L-(l,n)}, \hat{s}_{l,n}, \mathbf{y}_{1:K}, \mathbf{A})}. \tag{11}$$

and

$$\bar{w}_{l,n}^{(i)} = \frac{w_{l,n}^{(i)}}{\sum_{i=1}^{I} w_{l,n}^{(i)}} \tag{12}$$

The importance density is chosen as a uniform density such that $q(s_{l,n} \mid \mathbf{s}_{1:L-(l,n)}, \hat{s}_{l,n}, \mathbf{y}_{1:K}, \mathbf{A}) = q(s_{l,n} \mid \hat{s}_{l,n}, \theta) = \mathscr{U}(s_{l,n} \mid \hat{s}_{l,n} - \theta, \hat{s}_{l,n} + \theta)$ where θ is a scalar positive number that determines a bound for $s_{l,n}$.

In particle methods, all of the particles (samples) are saved in the memory and they are passed the next state. This procedure becomes more expensive for 2D images because each pixel has eight first order neighbors. For example, if we consider only the four past (causal) neighbors in the scanning direction, we must keep in memory all of the particles of top row pixels. Instead of keeping this amount of particles, we only keep their point estimates. This is the reason why we name the algorithm iterated posterior point estimate. The contribution of the point estimates of the neighboring pixels appears in prior densities which are the spatial transition densities. The accuracy of the particles, produced from a proposal density, is confirmed by the transition and the observation densities.

For estimating mixing matrix, we use the MSE estimate. Using (5), the MSE estimate of an element of $\mathbf{A}$ is written such that

$$\begin{aligned} \hat{a}_{k,l} &= \int a_{k,l} p(a_{k,l} \mid \mathbf{A}_{-(k,l)}, \mathbf{y}_{1:K}, \mathbf{s}_{1:L}) da_{k,l} \\ &= \int a_{k,l} p(\mathbf{y}_{1:K} \mid \mathbf{s}_{1:L}, a_{k,l}, \mathbf{A}_{-(k,l)}) da_{k,l} \end{aligned} \tag{13}$$

The likelihood of $a_{k,l}$ in the second integral is a Gaussian so the MSE estimate of $a_{k,l}$ can be simply found by the maximum likelihood method. But in this study, we also find it

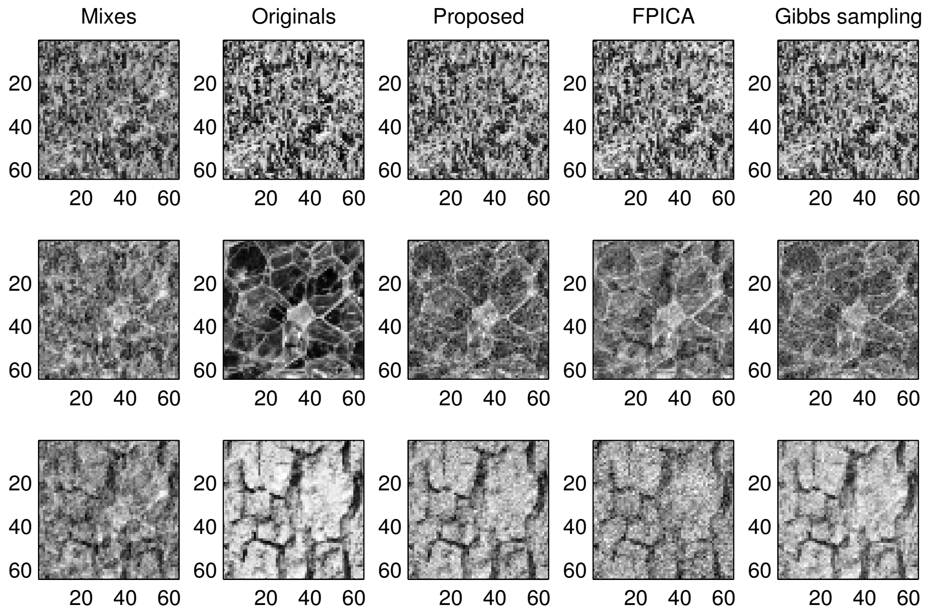

Figure 1: Simulation results of texture images under 40 dB SNR. First column, mixed images; second column originals; third column proposed; fourth column FPICA; fifth column Gibbs sampling.

using Monte Carlo integration to present a completely MC technique for using its benefits. The approximate integral is calculated such that

$$\hat{a}_{k,l} = \frac{1}{J}\sum_{j=1}^{J} a_{k,l}^{j} \tag{14}$$

where samples $a_{k,l}^{j}$ are drawn from the density $p(\mathbf{y}_{1:K}|\mathbf{s}_{1:L}, a_{k,l}, \mathbf{A}_{-(k,l)})$.

The algorithm is presented in Table 1.

4. SIMULATION RESULTS

The mixed images, which are formed by using the source images in the second column of Fig. 1, a linear mixing matrix and additive Gaussian noise, are shown in first column of Fig. 1. In this experiment, the SNR is 40 dB. The optimal power of robust error function $|.|^p$ is taken as $p = 1.8$ using an exhaustive search over a range of p values and the step size of the steepest descent in (8) is set at $\mu = 0.8$ after a few experimental runs. The number of particles on the mixing coefficients in Table 1, J is taken as 2. PSIR is checked at every 500 iteration and if the final PSIR is less than the PSIR of the previous one, the iterations are stopped.

In Table 4, Peak Signal-to-Interference Ratios (PSIR) comparison results of FPICA [10], ICM, Gibbs sampling (GS) and the proposed method are given. The outcomes of the FPICA, GS and the proposed algorithms can be visually assessed in Fig. 1. The result of the proposed method and that of the Gibbs sampling are very close to each other. But the Gibbs sampling takes a longer time to yield a result. In Table 4, the execution times are compared under different number of particles in the proposed method. The results in Fig. 1 are obtained with 64 particles. The first row of the Table 4 represents the Gibbs sampling values. The proposed method is two to three times faster than Gibbs sampling for small number of particles. If the number of particles is increased, our algorithm slows down. In the case of 128 particles, the best results is obtained but the speed is less than Gibbs sampling. Half a dB PSIR gain may be sacrificed to improve the time of convergence. In this case, 16 particles see to sufficient and one economizes 80 minutes in time.

I	PSIR	one iteration	T	total
GS	24.57	0.2267	60000	226.72
ICM	22.77	0.0432	80	0.0575
FPICA	22.28	0.0202	36	0.0121
2	23.29	0.8096	13900	188.65
4	23.85	0.8411	6220	87.19
8	23.89	0.8990	5450	81.66
16	24.09	1.0204	4810	81.80
32	24.29	1.2423	5100	105.59
64	24.56	1.7595	5700	167.15
128	24.62	2.6923	8900	399.35

Table 2: The PSIRs (dB) values according to number of particles I. Third column: duration of single iteration in seconds; fourth column: number of iteration; fifth column: total duration in minutes.

The PSIR of the third source as a function of the iterations are shown in Fig. 2 for 16, 32 and 64 particles. It is seen from figure that the optimum number of particles can be found between 32 and 64 particles. The variances of the PSIRs calculated using last 500 iterations are also given in Fig. 2. The variance of the PSIR can be used as a stopping criterion, for example $var(PSIR) < 0.1$. We finally show the evolution of posterior distribution from priors and the likelihood in Fig. 3.

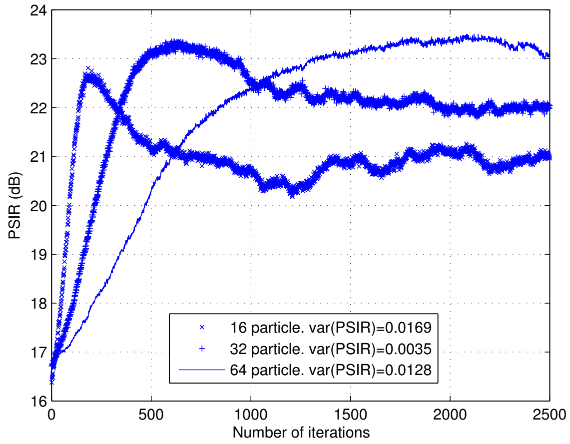

Figure 2: The PSIR of the third source as a function of the iteration number. The var(PSIR), shown as thickness of the curves, is computed using the last 500 PSIRs.

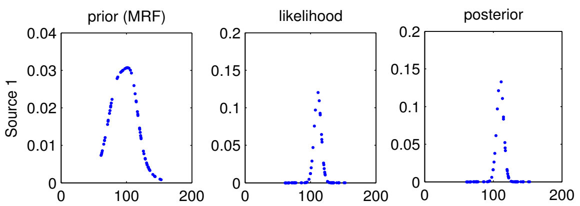

Figure 3: Sampled pdfs of a pixel in source 1. The columns correspond to the prior, the likelihood and the posterior pdfs.

5. CONCLUSION

In this study, we have introduced a new method for Bayesian image source separation. This method minimizes the expected value of some robust error function, and it resorts to particle methods to calculate stochastic integrals. For example, iterative estimation by steepest descent has a stochastic integral, which is conveniently solved by the particle method. Both the convergence time of the algorithm and the resulting PSIR figures increase monotonically with the number of particles. The SIR gain has diminishing returns beyond 64 particles. For large number of particles, the computation time increases and beyond 64 particles it exceeds that of Gibbs. Although the 64-particle operation point seems to be a good compromise between complexity and performance, we believe that both the performance and the speed of the algorithm can still be improved by choosing a more appropriate proposal density and adjusting more cleverly the step size. To implement a parallel sampling scheme by choosing a proper prior density instead of Gibbs distribution, convergence time can be reduced.

The proposed particle-based source separation method is applicable to diverse fields. We plan to pursue a realistic application problem, that of astrophysical image source separation [1].

In conclusion, the proposed method forms a niche source-separation solution that stands between Gibbs sampling and the parametric density fitting methods. Its advantage consists in the fact that the pdf of pixels are first estimated instead of directly the pixel themselves. Thus we work with a richer set of information, that is, the a posteriori pdf of pixels, and this can give rise to diverse estimators depending on the cost function used.

REFERENCES

[1] E. E. Kuruoglu, A. Tonazzini, and L. Bianchi, “Source separation in noisy astrophysical images modelled by Markov random fields,” in *Int. Conf. on Image Proc. ICIP'04*, Oct., 2004, pp. 24–27.

[2] A. Tonazzini, L. Bedini, and E. Salerno “A Markov model for blind image separation by a mean-filed EM algorithm,” *IEEE Trans. on Image Processing*, vol. 15, pp. 473–482, Feb. 2006.

[3] K. Kayabol, B. Sankur, and E. E. Kuruoglu, “Source separation in images via MRFs with variational approximation,” in *European Sig. Process. Conf. EUSIPCO'07*, Poznan, Sept. 2007, pp. 423-427.

[4] D. J. C. Mackay, “Introduction to Monte Carlo methods,” *Learning In Graphical Models*, MIT Press, pp. 175-204, 1999.

[5] D. Erdogmus, and J. C. Principe “Generalized information potential criterion for adaptive system training,” *IEEE Trans. on Neural Networks*, vol. 13, no.5, pp. 1035–1044, Sept. 2002.

[6] A. Doucet, S. Godsill, and C. Andrieu, “On sequential Monte Carlo sampling methods for Bayesian filtering,” *Statistics and Computing*, vol. 10, no. 3, pp. 197-208, 2000.

[7] M. Costagli, and, E. E. Kuruoglu “Image separation using particle filters, *Digital Signal Processing*, vol. 17, no. 5, pp. 935-946. Special issue Bayesian Source Separation. Elsevier, 2007.

[8] K. Nittono, and T. Kamakura, “On the use of particle filters for Bayesian image restoration,” in *Conf. on Computational Statistics, Compstat'02*, Berlin, 2002.

[9] Y. Zhai, M. Yeary, V. deBrunner, J. P. Havlicek, and O. Alkhouli, “Image restoration using a hybrid combination of particle filtering and wavelet denoising,” in *Int. Conf. on Image Proc. ICIP'05*, vol. 3, Sept. 2005, pp. 876-879.

[10] A. Hyvarinen, and E. Oja, “Fast fixed-point algorithm for independent component analysis,” *Neural Computation*, vol. 9, no. 7, pp. 1483-1492, 1997.