Bayesian Image Processing

Ali Mohammad-Djafari

Laboratoire des Signaux et Systèmes,
Unité mixte de recherche 8506 (CNRS-Supélec-UPS)
Supélec, Plateau de Moulon, 3 rue Joliot Curie, 91192 Gif-sur-Yvette, France.
djafari@lss.supelec.fr

ABSTRACT *Many image processing problems can be presented as inverse problems by modeling the relation of the observed image to the unknown desired features explicitly. Some of these problems are naturally presented as inverse problems such as restoration of blurred images (deconvolution) or image reconstruction in computed tomography. For some others, we need to translate the original problem as an inverse one. For example, image de-noising, image segmentation or even image compression can also be presented as inverse problems. The main advantage of doing so is that we can then use probabilistic modeling of the images and use the Bayesian estimation approach to propose new methods for them. In this paper, we present a very general forward modeling for the observations and a very general probabilistic modeling of images through a hidden Markov modeling (HMM) which can be used as the main basis for many image processing problems such as: 1) simple or multi channel image restoration, 2) simple or joint image segmentation, 3) multi-sensor data and image fusion and 4) Principal Component Analysis (PCA), Factor Analysis (FA), Independent Component Analysis (ICA) and blind source separation.*

KEYWORDS Inverse problems, Image Restoration, Image reconstruction, Image segmentation, Image registration and Multi-sensor image fusion, 3D image reconstruction, Tomography, Shape from shadows,

1 Introduction

A great number of image processing problems can be presented as inverse problems. The first step for this purpose is to model the relation of the observed image $g(\boldsymbol{r})$ to the unknown desired feature $f(\boldsymbol{r})$ explicitly. A very general form for such a relation is

$$g(\boldsymbol{r}) = [\mathcal{H}f](\boldsymbol{r}) + \epsilon(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{R} \tag{1}$$

where $\boldsymbol{r}$ represents the pixel position, $\mathcal{R}$ represents the whole surface of the observed images, $\mathcal{H}$ is an operator representing the forward problem and $\epsilon(\boldsymbol{r})$ represents the errors (modeling uncertainty and observation errors). When the operator $\mathcal{H}$ is linear we can write

$$g(\boldsymbol{r}) = \int_{\mathcal{R}'} f(\boldsymbol{r}')h(\boldsymbol{r}, \boldsymbol{r}')\,\mathrm{d}\boldsymbol{r}' + \epsilon(\boldsymbol{r}) \tag{2}$$

and when $h(\boldsymbol{r}, \boldsymbol{r}')$ is translation invariant, the relation becomes a convolution

$$g(\boldsymbol{r}) = \int_{\mathcal{R}'} f(\boldsymbol{r}')h(\boldsymbol{r}, \boldsymbol{r}')\boldsymbol{r}_{\cdot}' + \epsilon(\boldsymbol{r}) = h(\boldsymbol{r}) * f(\boldsymbol{r}) + \epsilon(\boldsymbol{r}) \tag{3}$$

and the corresponding inverse problem becomes image deblurring. When this relation is discretized, we can write it in a vector-matrix form

$$\boldsymbol{g} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{\epsilon} \tag{4}$$

where $\boldsymbol{g} = \{g(\boldsymbol{r}),\ \boldsymbol{r} \in \mathcal{R}\}$, $\boldsymbol{f} = \{f(\boldsymbol{r}),\ \boldsymbol{r} \in \mathcal{R}\}$ and $\boldsymbol{\epsilon} = \{\epsilon(\boldsymbol{r}),\ \boldsymbol{r} \in \mathcal{R}\}$ are vectors containing respectively the observed blurred image pixel values, the unknown original image pixel values and the observation errors and $\boldsymbol{H}$ is a huge dimensional matrix whose elements are defined from the system response function $h(\boldsymbol{r}, \boldsymbol{r}')$.

In case of a multi sensor observation data, a more general model is

$$\boldsymbol{g}_i = \boldsymbol{H}_i\boldsymbol{f}_i + \boldsymbol{\epsilon}_i, \quad i = 1, \cdots, M \tag{5}$$

and finally, we propose two more general models:

- General multi input multi output (MIMO) system

$$\boldsymbol{g}_i = \sum_{j=1}^{N} \boldsymbol{H}_{ij}\boldsymbol{f}_j + \boldsymbol{\epsilon}_i, \quad i = 1, \cdots, M \tag{6}$$

  where $\boldsymbol{H}_{ij}$ are assumed to be known, and

- General unknown mixing gain MIMO system

$$\boldsymbol{g}_i = \sum_{j=1}^{N} A_{ij}\boldsymbol{H}_j\boldsymbol{f}_j + \boldsymbol{\epsilon}_i, \quad i = 1, \cdots, M \tag{7}$$

  where $\boldsymbol{A} = \{A_{ij}, i = 1, \cdots, M,\ j = 1, \cdots, N\}$ is a known or unknown mixing matrix.

As we will see in the following, this forward modeling associated with a hidden Markov modeling (HMM) for the unknown images $\boldsymbol{f}_j$ can be used for modeling many image processing problems such as image de-noising or image segmentation, ($M = N = 1$ and $\boldsymbol{A} = 1, \boldsymbol{H} = 1$), single or multi channel image restoration ($M = N$ and $\boldsymbol{A} = 1$ and $\boldsymbol{H}_j$ known), image fusion ($M > 2$ and $N = 1$, $\boldsymbol{A}$ known) or PCA, FA, ICA and BSS ($M \neq N$ and $\boldsymbol{A}$ unknown).

2 Bayesian estimation framework

To illustrate the basics of the Bayesian estimation framework, we consider the general unknown mixing gain MIMO system (eq. 7) of the previous section where we assume that $\boldsymbol{H}_j$ and $\boldsymbol{A}$ are known. In what follows, we use the following notations: $\underline{\boldsymbol{g}} = \{\boldsymbol{g}_i, i = 1, \cdots, M\}$ and $\underline{\boldsymbol{f}} = \{\boldsymbol{f}_j, j = 1, \cdots, N\}$.

In a general Bayesian estimation framework, the forward model is used to define the likelihood function $p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}, \underline{\boldsymbol{\theta}}_1)$ and we have to translate our prior knowledge about the unknowns $\underline{\boldsymbol{f}}$ through a prior probability law $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{\theta}}_2)$ and then use the Bayes rule to find an expression for $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}})$

$$p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}}) \propto p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}, \underline{\boldsymbol{\theta}}_1)\, p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{\theta}}_2) \tag{8}$$

where $\underline{\boldsymbol{\theta}} = (\underline{\boldsymbol{\theta}}_1, \underline{\boldsymbol{\theta}}_2)$ represents all the hyperparameters (parameters of the likelihood and priors) of the problem.

When the expression of $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}})$ is obtained, we can use it to define any estimates for $\underline{\boldsymbol{f}}$. Two usual estimators are the maximum a posteriori (MAP) $\widehat{\underline{\boldsymbol{f}}} = \arg\max_{\underline{\boldsymbol{f}}}\{p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}})\}$ and the Mean Square Error (MSE) estimator which corresponds to the posterior mean $\widehat{\underline{\boldsymbol{f}}} = \int \underline{\boldsymbol{f}}\, p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}}) \underline{\boldsymbol{f}}$. Unfortunately only for the linear problems and the Gaussian laws where $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}})$ is also Gaussian we have analytical solutions for these two estimators. For almost all other cases, the first one needs an optimization algorithm and the second an integration one. For example, the relaxation methods can be used for the optimization and the MCMC algorithms can be used for expectation computations. Another difficult point is that the expressions of $p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}, \underline{\boldsymbol{\theta}}_1)$ and $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{\theta}}_2)$ and thus the expression of $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}})$ depend on the hyperparameters $\underline{\boldsymbol{\theta}}$ which, in practical applications, have also to be estimated either in a supervised way using the training data or in an unsupervised way. In both cases, we need also to translate our prior knowledge on them through a prior probability $p(\underline{\boldsymbol{\theta}})$. Thus, one of the main steps in any inversion method for any inverse problem is modeling the unknowns. In probabilistic methods and in particular in the Bayesian approach, this step becomes the assignment of the probability law $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{\theta}}_1)$. This point, as well as the assignment of $p(\underline{\boldsymbol{\theta}})$, are discussed the next two subsections.

2.1 HMM modeling of images

In general, any image $f_j(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}$ is composed of a finite set K_j of homogeneous regions R_{j_k} with given labels $z_j(\boldsymbol{r}) = k, k = 1, \cdots, K_j$ such that $R_{j_k} = \{\boldsymbol{r} \ : \ z_j(\boldsymbol{r}) = k\}$, $\mathcal{R}_j = \cup_k R_{j_k}$ and the corresponding pixel values $\boldsymbol{f}_{j_k} = \{f_j(\boldsymbol{r}) \ : \ \boldsymbol{r} \in R_{j_k}\}$ and $\boldsymbol{f}_j = \cup_k \boldsymbol{f}_{j_k}$. The Hidden Markov modeling (HMM) is a very general and efficient way to model appropriately such images. The main idea is to assume that all the pixel values $\boldsymbol{f}_{j_k} = \{f_j(\boldsymbol{r}), \boldsymbol{r} \in R_{j_k}\}$ of

a homogeneous region k follow a given probability law, for example a Gaussian $\mathcal{N}(m_{j_k}\mathbf{1}, \mathbf{\Sigma}_{j_k})$ where $\mathbf{1}$ is a generic vector of ones of the size n_{j_k} the number of pixels in region k.

In the following, we consider two cases:

- The pixels in a given region are assumed iid:

$$p(f_j(\boldsymbol{r})|z_j(\boldsymbol{r}) = k) = \mathcal{N}(m_{j_k}, \sigma^2_{j_k}), \quad k = 1, \cdots, K_j \tag{9}$$

  and thus

$$p(\boldsymbol{f}_{j_k}|z_j(\boldsymbol{r}) = k) = p(f_j(\boldsymbol{r}), \boldsymbol{r} \in R_{j_k}) = \mathcal{N}(m_{j_k}\mathbf{1}, \sigma^2_{j_k}\boldsymbol{I}) \tag{10}$$

- The pixels in a given region are assumed to be locally dependent:

$$p(\boldsymbol{f}_{j_k}|z_j(\boldsymbol{r}) = k) = p(f_j(\boldsymbol{r}), \boldsymbol{r} \in R_{j_k}) = \mathcal{N}(m_{j_k}\mathbf{1}, \mathbf{\Sigma}_{j_k}) \tag{11}$$

  where $\mathbf{\Sigma}_{j_k}$ is an appropriate covariance matrix.

In both cases, the pixels in different regions are assumed to be independent:

$$p(\boldsymbol{f}_j) = \prod_{k=1}^{K_j} p(\boldsymbol{f}_{j_k}) = \prod_{k=1}^{K_j} \mathcal{N}(m_{j_k}\mathbf{1}, \mathbf{\Sigma}_{j_k}). \tag{12}$$

2.2 Modeling the labels

Noting that all the models (9), (10) and (11) are conditioned on the value of $z_j(\boldsymbol{r}) = k$, they can be rewritten in the following general form

$$p(\boldsymbol{f}_{j_k}) = \sum_k P(z_j(\boldsymbol{r}) = k)\, \mathcal{N}(m_{j_k}, \mathbf{\Sigma}_{j_k}) \tag{13}$$

where either $\mathbf{\Sigma}_{j_k}$ is a diagonal matrix $\mathbf{\Sigma}_{j_k} = \sigma^2_{j_k}\boldsymbol{I}$ or not. Now, we need also to model the vector variables $\boldsymbol{z}_j = \{z_j(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}\}$. Here also, we can consider two cases:

- Independent Gaussian Mixture model (IGM), where $\{z_j(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}\}$ are assumed to be independent and

$$P(z_j(\boldsymbol{r}) = k) = p_k, \quad \text{with} \quad \sum_k p_k = 1 \ \text{ and } \ p(\boldsymbol{z}_j) = \prod_k p_k \tag{14}$$

- Contextual Gaussian Mixture model (CGM), where $\boldsymbol{z}_j = \{z_j(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}\}$ are assumed to be Markovian

$$p(\boldsymbol{z}_j) \propto \exp\left[\alpha \sum_{\boldsymbol{r}\in\mathcal{R}} \sum_{\boldsymbol{s}\in\mathcal{V}(\boldsymbol{r})} \delta(z_j(\boldsymbol{r}) - z_j(\boldsymbol{s}))\right] \tag{15}$$

  which is the Potts Markov random field (PMRF). The parameter α controls the mean value of the regions' sizes.

2.3 Hyperparameters prior law

The final point before obtaining an expression for the posterior probability law of all the unknowns, i.e, $p(\underline{\boldsymbol{f}}, \boldsymbol{\theta}|\underline{\boldsymbol{g}})$ is to assign a prior probability law $p(\underline{\boldsymbol{\theta}})$ to the hyperparameters $\underline{\boldsymbol{\theta}}$. Even if this point has been one of the main discussing points between Bayesian and classical statistical research community, and still there are many open problems, we choose here to use the conjugate priors for simplicity. The conjugate priors have at least two advantages: 1) they can be considered as a particular family of a differential geometry based family of priors [1,2] and 2) they are easy to use because the prior and the posterior probability laws stay in the same family. In our case, we need to assign prior probability laws to the means m_{j_k}, to the variances $\sigma^2_{j_k}$ or to the covariance matrices $\boldsymbol{\Sigma}_{j_k}$ and also to the covariance matrices of the noises $\boldsymbol{\epsilon}_i$ of the likelihood functions. The conjugate priors for the means m_{j_k} are in general the Gaussians $\mathcal{N}(m_{j_{k0}}, \sigma^2_{j_{k0}})$, those of variances $\sigma^2_{j_k}$ are the inverse Gamma's $\mathcal{IG}(\alpha_0, \beta_0)$ and those for the covariance matrices $\boldsymbol{\Sigma}_{j_k}$ are the inverse Wishart's $\mathcal{IW}(\alpha_0, \boldsymbol{\Lambda}_0)$.

2.4 Expressions of likelihood, prior and posterior laws

We now have all the elements for writing the expressions of the posterior laws. We are going to summarizes them here:

- Likelihood: $p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}, \underline{\boldsymbol{\theta}}) = \prod_{i=1}^{M} p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}, \boldsymbol{\Sigma}_{\epsilon i}) = \prod_{i=1}^{M} \mathcal{N}(\underline{\boldsymbol{g}} - \underline{\boldsymbol{f}}, \boldsymbol{\Sigma}_{\epsilon i})$
  where we assumed that the noises $\boldsymbol{\epsilon}_i$ are independent, centered and Gaussian with covariance matrices $\boldsymbol{\Sigma}_{\epsilon i}$ which, hereafter, are also assumed to be diagonal $\boldsymbol{\Sigma}_{\epsilon i} = \sigma^2_{\epsilon_i} \boldsymbol{I}$.
- HMM for the images: $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{z}}, \underline{\boldsymbol{\theta}}) = \prod_{j=1}^{N} p(\boldsymbol{f}_j|\boldsymbol{z}_j, \boldsymbol{m}_j, \boldsymbol{\Sigma}_j)$
  where we used $\underline{\boldsymbol{z}} = \{\boldsymbol{z}_j, j = 1, \cdots, N\}$ and where we assumed that $\boldsymbol{f}_j|\boldsymbol{z}_j$ are independent.
- PMRF for the labels: $p(\underline{\boldsymbol{z}}) \propto \prod_{j=1}^{N} \exp\left[\alpha \sum_{\boldsymbol{r}\in\mathcal{R}} \sum_{\boldsymbol{s}\in\mathcal{V}(\boldsymbol{r})} \delta(z_j(\boldsymbol{r}) - z_j(\boldsymbol{s}))\right]$
  where we used the simplified notation $p(\boldsymbol{z}_j) = P(Z_j(\boldsymbol{r}) = z(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R})$ and where we assumed $\{\boldsymbol{z}_j, j = 1, \cdots, N\}$ are independent.

- Conjugate priors for the hyperparameters:
  $p(m_{j_k}) = \mathcal{N}(m_{j_{k0}}, \sigma^2_{j_{k0}})$, $p(\sigma^2_{j_k}) = \mathcal{IG}(\alpha_{j0}, \beta_{j0})$,
  $p(\mathbf{\Sigma}_{j_k}) = \mathcal{IW}(\alpha_{j0}, \Lambda_{j0})$, $p(\sigma_{\epsilon i}) = \mathcal{IG}(\alpha_{i0}, \beta_{i0})$.
- Joint posterior law of $\underline{\boldsymbol{f}}$, $\underline{\boldsymbol{z}}$ and $\underline{\boldsymbol{\theta}}$

$$p(\underline{\boldsymbol{f}}, \underline{\boldsymbol{z}}, \underline{\boldsymbol{\theta}}|\underline{\boldsymbol{g}}) \propto p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}, \boldsymbol{\theta}_1)\, p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{z}}, \boldsymbol{\theta}_2)\, p(\underline{\boldsymbol{z}}|\boldsymbol{\theta}_2)\, p(\underline{\boldsymbol{\theta}})$$

The general MCMC algorithm we propose to estimate $\underline{\boldsymbol{f}}$, $\underline{\boldsymbol{z}}$ and $\underline{\boldsymbol{\theta}}$ by their MSE estimators is a Gibbs sampling scheme where we first separate the unknowns in two sets $p(\underline{\boldsymbol{f}}, \underline{\boldsymbol{z}}|\underline{\boldsymbol{g}}, \underline{\boldsymbol{\theta}})$ and $p(\underline{\boldsymbol{\theta}}|\underline{\boldsymbol{f}}, \underline{\boldsymbol{z}}, \underline{\boldsymbol{g}})$. Then, we separate again the first set in two subsets $p(\underline{\boldsymbol{f}}|\underline{\boldsymbol{z}}, \underline{\boldsymbol{\theta}}, \underline{\boldsymbol{g}})$ and $p(\underline{\boldsymbol{z}}|\underline{\boldsymbol{\theta}}, \underline{\boldsymbol{g}})$. Finally, when possible, using the separability along the channels, separate these two last terms in $p(\boldsymbol{f}_j|\boldsymbol{z}_j, \boldsymbol{\theta}_j, \boldsymbol{g}_j)$ and $p(\boldsymbol{z}_j|\boldsymbol{\theta}_j, \boldsymbol{g}_j)$. The general scheme is then, using these expressions, to generates samples $\underline{\boldsymbol{f}}^{(n)}, \underline{\boldsymbol{z}}^{(n)}, \underline{\boldsymbol{\theta}}^{(n)}$ from the joint posterior law $p(\underline{\boldsymbol{f}}, \underline{\boldsymbol{z}}, \underline{\boldsymbol{\theta}}|\underline{\boldsymbol{g}})$ and after the convergence of the Gibbs samplers, to compute their mean and to use them as the posterior estimates.

In the following section we examine some particular cases.

3 Particular examples

3.1 Single channel image restoration

The forward model and the priors for this case can be summarized as follows:

$$\begin{cases}
g(\boldsymbol{r}) = h(\boldsymbol{r}) * f(\boldsymbol{r}) + \epsilon(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{R} \;\text{ or }\; \boldsymbol{g} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{\epsilon} \\
p(\boldsymbol{g}|\boldsymbol{f}) = \mathcal{N}(\boldsymbol{H}\boldsymbol{f}, \mathbf{\Sigma}_{\boldsymbol{\epsilon}}) \;\text{ with }\; \mathbf{\Sigma}_{\boldsymbol{\epsilon}} = {\sigma_\epsilon}^2 \boldsymbol{I} \\
p(f(\boldsymbol{r})|z(\boldsymbol{r}) = k) = \mathcal{N}(m_k, \sigma_k^2), \quad k = 1, \cdots, K \\
p(\boldsymbol{f}_k|z(\boldsymbol{r}) = k) = \mathcal{N}(m_k \mathbf{1}_k, \mathbf{\Sigma}_k) \;\text{ with }\; \mathbf{\Sigma}_k = \sigma_k^2 \boldsymbol{I}_k \\
p(\boldsymbol{z}) = p(z(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}) \propto \exp\left[\alpha \sum_{\boldsymbol{r}\in\mathcal{R}} \sum_{s\in\mathcal{V}(\boldsymbol{r})} \delta(z(\boldsymbol{r}) - z(s))\right] \\
p(\boldsymbol{f}|\boldsymbol{z}) = \prod_k \mathcal{N}(m_k \mathbf{1}_k, \mathbf{\Sigma}_k) = \mathcal{N}(\boldsymbol{m}_z, \mathbf{\Sigma}_z) \;\text{ with} \\
\boldsymbol{m}_z = [m_1 \mathbf{1}'_1, \cdots, m_K \mathbf{1}'_K]' \;\text{ and }\; \mathbf{\Sigma}_z = \text{diag}\,[\Sigma_1, \cdots, \Sigma_K] \\
p(m_k) = \mathcal{N}(m_{k0}, \sigma^2_{k0}), \quad p(\sigma_k^2) = \mathcal{IG}(\alpha_{k0}, \beta_{k0}), \quad p({\sigma_\epsilon}^2) = \mathcal{IG}(\alpha_0^\epsilon, \beta_0^\epsilon)
\end{cases}$$

and the posterior probability laws are:

$$\begin{cases}
p(\boldsymbol{f}|\boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{g}) = \mathcal{N}(\widehat{\boldsymbol{f}}, \widehat{\mathbf{\Sigma}}) \;\text{ with} \\
\widehat{\mathbf{\Sigma}} = (\boldsymbol{H}^t {\mathbf{\Sigma}_{\boldsymbol{\epsilon}}}^{-1} \boldsymbol{H} + {\mathbf{\Sigma}_z}^{-1})^{-1} \;\text{ and }\; \widehat{\boldsymbol{f}} = \widehat{\mathbf{\Sigma}}\left({\mathbf{\Sigma}_{\boldsymbol{\epsilon}}}^{-1} \boldsymbol{H}^t \boldsymbol{g} + {\mathbf{\Sigma}_z}^{-1} \boldsymbol{m}_z\right) \\
p(\boldsymbol{z}|\boldsymbol{g}, \boldsymbol{\theta}) \propto p(\boldsymbol{g}|\boldsymbol{z}, \boldsymbol{\theta})\, p(\boldsymbol{z}) \;\text{ with} \\
p(\boldsymbol{g}|\boldsymbol{z}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{m}_z, \mathbf{\Sigma}_{\boldsymbol{g}}) \;\text{ with }\; \mathbf{\Sigma}_{\boldsymbol{g}} = \boldsymbol{H}\mathbf{\Sigma}_z\boldsymbol{H}^t + \mathbf{\Sigma}_{\boldsymbol{\epsilon}} \\
p(m_k|\boldsymbol{z}, \boldsymbol{f}) = \mathcal{N}(\mu_k, v_k^2) \text{ with } v_k^2 = \left(\frac{n_k}{\sigma_k^2} + \frac{1}{{\sigma_\epsilon}^2}\right)^{-1} \text{and } \mu_k = v_k^2 \left(\frac{m_{k0}}{\sigma^2_{k0}} + \frac{n_k \bar{f}_k}{{\sigma_\epsilon}^2}\right) \\
p(\sigma_k^2|\boldsymbol{f}, \boldsymbol{z}) = \mathcal{IG}(\alpha_k, \beta_k) \text{ with } \alpha_k = \alpha_{k0} + \frac{n_k}{2} \text{ and } \beta_k = \beta_{k0} + \frac{n_k \bar{s}_k}{2} \\
\text{where } \bar{f}_k = \frac{1}{n_k} \sum_{\boldsymbol{r}\in R_k} f_i(\boldsymbol{r}) \text{ and } \bar{s}_k = \sum_{\boldsymbol{r}\in R_k} (f(\boldsymbol{r}) - m_k)^2 \\
p({\sigma_\epsilon}^2|\boldsymbol{f}, \boldsymbol{g}) = \mathcal{IG}(\alpha^\epsilon, \beta^\epsilon) \text{ with } \alpha^\epsilon = \frac{n}{2} + \alpha_0^\epsilon \;\text{ and }\; \beta^\epsilon = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\boldsymbol{f}\|^2 + \beta_0^\epsilon \\
n_k = \text{number of pixels in } R_k \;\text{ and }\; n = \text{total number of pixels.}
\end{cases}$$

For an application see [3].

3.2 Registered images fusion and joint segmentation

Here, each observed image $g_i(\boldsymbol{r})$ (or equivalently $\boldsymbol{g}_i$) is assumed to be a noisy version of the unobserved real image $f_i(\boldsymbol{r})$ (or equivalently $\boldsymbol{f}_i$) and all the unobserved real images $f_i(\boldsymbol{r}),\ \ i = 1, \cdots, M$ are assumed to have a common segmentation $z(\boldsymbol{r})$ (or equivalently $\boldsymbol{z}$) which is modeled by a discrete value Potts Random Markov Field (PRMF):

$$
\left\{
\begin{array}{l}
g_i(\boldsymbol{r}) = f_i(\boldsymbol{r}) + \epsilon_i(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{R}, \ \text{ or } \ \boldsymbol{g}_i = \boldsymbol{f}_i + \boldsymbol{\epsilon}_i, \quad i = 1, \cdots, M \\
p(\underline{\boldsymbol{g}}|\underline{\boldsymbol{f}}) = \prod_i p(\boldsymbol{g}_i|\boldsymbol{f}_i) \ \text{ with} \\
p(\boldsymbol{g}_i|\boldsymbol{f}_i) = \mathcal{N}(\boldsymbol{f}_i, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}i}) \ \text{ with } \ \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}i} = \sigma_{\epsilon_i}^2 \boldsymbol{I} \\
p(f_i(\boldsymbol{r})|z(\boldsymbol{r}) = k) = \mathcal{N}(m_{ik}, \sigma_{i\,k}^2), \quad k = 1, \cdots, K \\
p(\boldsymbol{f}_{ik}|z(\boldsymbol{r}) = k) = \mathcal{N}(m_{ik}\mathbf{1}_k, \boldsymbol{\Sigma}_{ik}) \ \text{ with } \ \boldsymbol{\Sigma}_{ik} = \sigma_{i\,k}^2 \boldsymbol{I}_k \\
p(\boldsymbol{z}) = p(z(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}) \propto \exp\left[\alpha \sum_{\boldsymbol{r}\in\mathcal{R}} \sum_{s\in\mathcal{V}(\boldsymbol{r})} \delta(z(\boldsymbol{r}) - z(s))\right] \\
p(\boldsymbol{f}_i|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{m}_{zi}, \boldsymbol{\Sigma}_{zi}) \ \text{ with} \\
\boldsymbol{m}_{zi} = [m_{i1}\mathbf{1}_1', \cdots, m_{iK}\mathbf{1}_K']' \ \text{ and } \ \boldsymbol{\Sigma}_{zi} = \text{diag}\,[\Sigma_{i1}, \cdots, \Sigma_{iK}] \\
p(m_{ik}) = \mathcal{N}(m_{ik0}, \sigma_{i\,k0}^2), \quad p(\sigma_{i\,k}^2) = \mathcal{IG}(\alpha_{i0}, \beta_{i0}), \quad p(\sigma_{\epsilon_i}^2) = \mathcal{IG}(\alpha_{i0}^{\epsilon}, \beta_{i0}^{\epsilon}) \\
p(\underline{\boldsymbol{f}}|\boldsymbol{z}) = \prod_i p(\boldsymbol{f}_i|\boldsymbol{z})
\end{array}
\right.
$$

and we have

$$
\left\{
\begin{array}{l}
p(\boldsymbol{f}_i|\boldsymbol{z}, \boldsymbol{\theta}_i, \boldsymbol{g}_i) = \mathcal{N}(\widehat{\boldsymbol{f}}_i, \widehat{\boldsymbol{\Sigma}}_i) \ \text{ with} \\
\widehat{\boldsymbol{\Sigma}}_i = (\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_i}^{-1} + \boldsymbol{\Sigma}_z{}^{-1})^{-1} \ \text{ and } \ \widehat{\boldsymbol{f}}_i = \widehat{\boldsymbol{\Sigma}}_i \left(\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_i}{}^{-1}\boldsymbol{g}_i + \boldsymbol{\Sigma}_{z_i}{}^{-1}\boldsymbol{m}_{zi}\right) \\
p(\boldsymbol{z}|\underline{\boldsymbol{g}}, \boldsymbol{\theta}) \propto \left(\prod_i p(\boldsymbol{g}_i|\boldsymbol{z}, \boldsymbol{\theta}_i)\right)\ p(z(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}) \ \text{ with} \\
p(\boldsymbol{g}_i|\boldsymbol{z}, \boldsymbol{\theta}_i) = \mathcal{N}(\boldsymbol{m}_{zi}, \boldsymbol{\Sigma}_{\boldsymbol{g}_i}) \ \text{ with } \ \boldsymbol{\Sigma}_{\boldsymbol{g}_i} = \boldsymbol{\Sigma}_{zi} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}i} \\
p(m_{ik}|\boldsymbol{f}_i, \boldsymbol{z}, \sigma_{i\,k}^2) = \mathcal{N}(\mu_{ik}, v_{ik}^2) \ \text{ with} \\
\mu_{ik} = v_{ik}^2 \left(\frac{m_{i0}}{\sigma_{i\,0}^2} + \frac{n_k \bar{f}_{ik}}{\sigma_{i\,k}^2}\right) \ \text{ and } \ v_{ik}^2 = \left(\frac{n_k}{\sigma_{i\,k}^2} + \frac{1}{\sigma_{i\,0}^2}\right)^{-1} \\
p(\sigma_{i\,k}^2|\boldsymbol{f}_i, \boldsymbol{z}) = \mathcal{IG}(\alpha_{ik}, \beta_{ik}) \text{ with } \alpha_{ik} = \alpha_{i0} + \frac{n_k}{2} \ \text{ and } \ \beta_{ik} = \beta_{i0} + \frac{\bar{s}_i}{2} \\
\text{where } \bar{f}_{ik} = \frac{1}{n_k}\sum_{\boldsymbol{r}\in R_k} f_i(\boldsymbol{r}) \text{ and } \bar{s}_i = \sum_{\boldsymbol{r}\in R_k} \left(f_i(\boldsymbol{r}) - m_{ik}\right)^2 \\
p(\sigma_{\epsilon_i}^2|\boldsymbol{f}_i, \boldsymbol{g}_i) = \mathcal{IG}(\alpha_i^{\epsilon}, \beta_i^{\epsilon}) \text{ with } \alpha_i^{\epsilon} = \frac{n}{2} + \alpha_{i0}^{\epsilon} \ \text{ and } \ \beta_i^{\epsilon} = \frac{1}{2}\|\boldsymbol{g}_i - \boldsymbol{f}_i\|^2 + \beta_{i0}^{\epsilon} \\
n_k = \text{number of pixels in } R_k, \quad n = \text{total number of pixels.}
\end{array}
\right.
$$

For more details on this model and its application in medical image fusion as well as in image fusion for security systems see [4].

3.3 Joint segmentation of hyper-spectral images

The proposed model is the same as the model of the previous section except for the last equation of the forward model which assumes that the pixels in similar regions of different images are independent. For hyper-spectral images, this hypothesis is not valid and we have to account for their correlations. This work is under consideration.

3.4 Segmentation of a video sequence of images

Here, we can not assume that all the images in the video sequence have the same segmentation labels. However, we may use the segmentation obtained in an image as an initialization for the segmentation of next image. For more details on this model and to see a typical result see [5].

3.5 Joint segmentation and separation of instantaneous mixed images

Here, the additional difficulty is that we also have to estimate the mixing matrix $\boldsymbol{A}$. For more details on this model and to see some typical result in joint segmentation and separation of images see [2].

4 Conclusion

In this paper we first showed that many image processing problems can be presented as inverse problems by modeling the relation of the observed image to the unknown desired features explicitly. Then, we presented a very general forward modeling for the observations and a very general probabilistic modeling of images through a hidden Markov modeling (HMM) which can be used as the main basis for many image processing problems such as: 1) simple or multi channel image restoration, 2) simple or joint image segmentation, 3) multi-sensor data and image fusion, 4) joint segmentation of color or hyperspectral images and v) joint blind source separation (BSS) and segmentation. Finally, we presented detailed forward models, prior and posterior probability law expressions for the implementation of MCMC algorithms for a few of those problems.

REFERENCES

[1] H. Snoussi and A. Mohammad-Djafari,*Information Geometry and Prior Selection,* Bayesian Inference and Maximum Entropy Methods, C. Williams, Ed. MaxEnt Workshops, Aug. 2002, pp. 307–327, Amer. Inst. Physics.

[2] H. Snoussi and A. Mohammad-Djafari,*Fast joint separation and segmentation of mixed images,* Journal of Electronic Imaging, 2003.

[3] A. Mohammad-Djafari,*Bayesian approach for inverse problems in optics, Presented SPIE, Optical Information Systems, aug. 2003, San Diego, USA.,*

[4] O. Féron and A. Mohammad-Djafari,*Image fusion and joint segmentation using an MCMC algorithm,* Submitted to J. of Electronic Imaging, April 2004.

[5] P. Brault and A. Mohammad-Djafari,*Bayesian Segmentation and Motion Estimation in Video Sequences using a Markov-Potts Model,* WSEAS04, 2004.