Hierarchical Markovian Models for Joint Classification, Segmentation and Data Reduction of Hyperspectral Images

Nadia BALI, Ali MOHAMMAD-DJAFARI and Adel MOHAMMADPOOR

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.

Abstract. Spectral classification, segmentation and data reduction are the three main problems in hyperspectral image analysis. In this paper we propose a Bayesian estimation approach which tries to give a solution for these three problems jointly. The data reduction problem is modeled as a blind sources separation (BSS) where the data are the m hyperspectral images and the sources are the $n < m$ images which must be mutually the most independent and piecewise homogeneous. To insure these properties, we propose a hierarchical model for the sources with a common hidden classification variable which is modelled via a Potts-Markov field. The joint Bayesian estimation of this hidden variable as well as the sources and the mixing matrix of the BSS problem gives a solution for all the three problems of spectra classification, segmentation and data reduction problems of hyperspectral images. A few simulation results are given to illustrate the performances of the proposed method and some comparison with other classical methods of PCA and ICA used for BSS.

1 Introduction

Hyperspectral images data are often represented either as a set of images $x_\omega(\boldsymbol{r})$ or as a set of spectra $x_{\boldsymbol{r}}(\omega)$ where $\omega \in \{1, \cdots, \Omega\}$ indexes the wavelength and $\boldsymbol{r} \in \mathcal{R}$ is a pixel position [1, 2, 3]. In both representations, the data are dependent in both spatial positions and in spectral bands. Classical methods of hyperspectral image analysis try either to classify the spectra $x_\omega(\boldsymbol{r})$ in K classes $\{s_k(\omega), k = 1, \cdots, K\}$ or to classify the images $x_\omega(\boldsymbol{r})$ in N classes $\{s_j(\boldsymbol{r}), j = 1, \cdots, N\}$, using in both cases, the classical classification methods such as distance based methods (like K-means) or probabilistic methods using the mixture of Gaussian (MoG) modeling of the data. However, these methods either neglect the spatial structure of the spectra or the spectral natures of the pixels along the wavelength bands.

If we consider the data as a set of spectra, then we want to write:

$$x_{\boldsymbol{r}}(\omega) = \sum_{k=1}^{K} A_{\boldsymbol{r},k}\ s_k(\omega) + \epsilon_{\boldsymbol{r}}(\omega), \tag{1}$$

where the $s_k(\omega)$ are the K spectral sources and each column of the mixing matrix $\boldsymbol{A}$ is in fact an image $A_k(\boldsymbol{r})$. The ideal case here would be to obtain an estimate

for $\boldsymbol{A}$ such that each column $A_k(\boldsymbol{r})$ represents an image where only non-zero values for the pixels in the regions which are associated to the spectrum $s_k(\omega)$. At the other hand, if we consider the data as a set of images $x_\omega(\boldsymbol{r})$, then we have:

$$x_\omega(\boldsymbol{r}) = \sum_{j=1}^{N} A_{\omega,j}\ s_j(\boldsymbol{r}) + \epsilon_\omega(\boldsymbol{r}), \tag{2}$$

where the sources $s_j(\boldsymbol{r})$ are the N source images and each column of the mixing matrix $\boldsymbol{A}$ in this case correspond to the spectrum $A_j(\omega)$. The ideal case here would be to obtain an estimate for the sources such that the pixels of each image $s_j(\boldsymbol{r})$ be non-zero only for the positions in the regions which are associated to the spectrum $A_j(\omega)$.

In this paper, we propose to consider the data reduction problem as a blind sources separation (BSS) and use a Bayesian estimation framework with a hierarchical model for the sources with a common hidden classification variable which is modelled via a Potts-Markov field. The joint estimation of this hidden variable as well as the sources and the mixing matrix of the BSS problem gives a solution for all the three problems of spectra classification, segmentation and data reduction problems of hyperspectral images.

2 Proposed data reduction model and method

2.1 Data reduction model

As explained in the introduction, we propose to consider the data reduction problem as in equation (2), written in vector form:

$$\boldsymbol{x}(\boldsymbol{r}) = \boldsymbol{A}\boldsymbol{s}(\boldsymbol{r}) + \boldsymbol{\epsilon}(\boldsymbol{r}) \tag{3}$$

where $\boldsymbol{x}(\boldsymbol{r}) = \{x_i(\boldsymbol{r}), i = 1, \cdots, m\}$ is the set of m observed mixed images (hyperspectral images), $\boldsymbol{A}$ the unknown mixing matrix of dimensions (m, n), $\boldsymbol{s}(\boldsymbol{r}) = \{s_j(\boldsymbol{r}), j = 1, \cdots, n\}$ the set of n unknown components (source images) and $\boldsymbol{\epsilon}(\boldsymbol{r}) = \{\epsilon_i(\boldsymbol{r}), i = 1, \cdots, m\}$ represents the errors. Now, if we note by $\underline{\boldsymbol{x}} = \{\boldsymbol{x}(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}\}$, $\underline{\boldsymbol{s}} = \{\boldsymbol{s}(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}\}$ and $\underline{\boldsymbol{\epsilon}} = \{\boldsymbol{\epsilon}(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}\}$,
then we can write

$$\underline{\boldsymbol{x}} = \boldsymbol{A}\underline{\boldsymbol{s}} + \boldsymbol{\epsilon} \tag{4}$$

In the following, we assume that the errors $\boldsymbol{\epsilon}(\boldsymbol{r})$ are centered, white, Gaussian with covariance matrix $\boldsymbol{\Sigma}_\epsilon = diag(\sigma^2_{\epsilon_1}, \cdots, \sigma^2_{\epsilon_m})$. This leads to

$$p(\underline{\boldsymbol{x}}|\underline{\boldsymbol{s}}, \boldsymbol{A}, \boldsymbol{\Sigma}_\epsilon) = \prod_{\boldsymbol{r}} \mathcal{N}(\boldsymbol{A}\boldsymbol{s}(\boldsymbol{r}), \boldsymbol{\Sigma}_\epsilon) \tag{5}$$

2.2 Sources model

As we mentioned in the introduction, we want to impose to all these sources $\boldsymbol{s}(\boldsymbol{r})$ to be piecewise homogeneous and share the same segmentation. This can

be achieved via the introduction of a discrete valued hidden variable $z(\boldsymbol{r})$ and by assuming the following:

$$p(s_j(\boldsymbol{r})|z(\boldsymbol{r})=k)) = \mathcal{N}(m_{j\,k}, \sigma^2_{j\,k}) \tag{6}$$

and

$$p(z(\boldsymbol{r}), \boldsymbol{r}\in\mathcal{R}) \propto \exp\left[\beta \sum_{\boldsymbol{r}\in\mathcal{R}} \sum_{\boldsymbol{r}'\in\mathcal{V}(\boldsymbol{r})} \delta(z(\boldsymbol{r})-z(\boldsymbol{r}'))\right] \tag{7}$$

$z(\boldsymbol{r})$ then will represents the common segmentation of the sources and the data. We also impose $m_{j\,k}=0$ if $j\neq k$ and $\sigma^2_{j\,k}=.001$ if $j\neq k$ which try to insure that each image $s_j(\boldsymbol{r})$ be composed of zeros everywhere except those regions associated with class k.

2.3 Data and sources hierarchical model

Combining the data and the sources models of the previous section, we obtain the following hierarchical model:

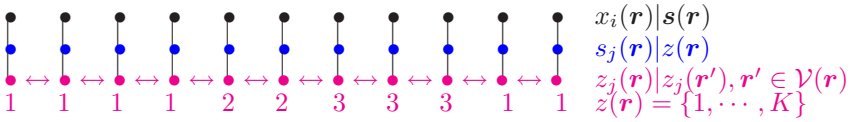

Fig. 1: **Proposed Model:** A hierarchical model for data $x_i(\boldsymbol{r})$ and sources $s_j(\boldsymbol{r})$ with hidden common classification and segmentation variables $z(\boldsymbol{r})$.

3 Bayesian estimation framework

The first step in the Bayesian estimation approach is to find expressions for the likelihood $p(\underline{\boldsymbol{x}}|\underline{\boldsymbol{s}}, \boldsymbol{A}, \boldsymbol{R}_\epsilon)$ and for the sources as we did in previous section. However, these expressions depend on some hyperparameters $\underline{\boldsymbol{\theta}}=\{\boldsymbol{\theta}_\epsilon, \boldsymbol{\theta}_s\}$ where $\boldsymbol{\theta}_\epsilon=\boldsymbol{R}_\epsilon$ and $\boldsymbol{\theta}_s=\{(m_{j\,k}, \sigma^2_{j\,k})\}$. So, we have to assign $p(\underline{\boldsymbol{\theta}})$ and also $p(\boldsymbol{A})$. In the following we use conjugate priors for all of them, i.e., Gaussian for the elements of $\boldsymbol{A}$, Gaussian for the means $m_{j\,k}$ and inverse Gamma for the variances $\sigma^2_{j\,k}$ as well as for the noise variances $\sigma_{\epsilon i}$.

When all these priors are appropriately assigned, we can obtain an expression for the posterior law

$$p(\underline{\boldsymbol{s}}, \boldsymbol{z}, \boldsymbol{A}, \underline{\boldsymbol{\theta}}|\underline{\boldsymbol{x}}) \propto p(\underline{\boldsymbol{x}}|\underline{\boldsymbol{s}}, \boldsymbol{A}, \boldsymbol{\theta}_\epsilon)\, p(\underline{\boldsymbol{s}}|\boldsymbol{z}, \boldsymbol{\theta}_s)\, p(\underline{\boldsymbol{\theta}}) \tag{8}$$

We can then use this posterior law to define an estimator such as Joint Maximum A Posteriori (JMAP) or the Posterior Means (PM). The first needs optimization algorithms and the second integration methods. Both are computationally demanding. Alternate optimization is generally used for the first while the MCMC techniques are used for the second. We propose here the following algorithm:

- First, integrate out $\underline{\boldsymbol{s}}$ to obtain $p(\boldsymbol{z}, \boldsymbol{A}, \underline{\boldsymbol{\theta}}|\underline{\boldsymbol{x}})$ and then estimate $\boldsymbol{z}$ by

$$\widehat{\boldsymbol{z}} \sim p(\boldsymbol{z}|\widehat{\boldsymbol{A}}, \widehat{\underline{\boldsymbol{\theta}}}, \underline{\boldsymbol{x}})$$

- Then, compute the posterior mean of the sources using $p(\underline{\boldsymbol{s}}|\widehat{\boldsymbol{z}}, \widehat{\boldsymbol{A}}, \widehat{\underline{\boldsymbol{\theta}}}, \underline{\boldsymbol{x}})$.
- Third estimate $\boldsymbol{A}$ and $\underline{\boldsymbol{\theta}}$ using $p(\boldsymbol{A}, \underline{\boldsymbol{\theta}}|\underline{\boldsymbol{s}}, \boldsymbol{z}, \underline{\boldsymbol{x}})$

In this algorithm, $\sim$ represents either *argmax* or *generate sample using* or still *compute the Mean Field Approximation (MFA)*. In the following, we give detail expressions of the different conditional laws as well as the details of the proposed algorithm.

4 Expressions of the a posteriori conditional laws

- $p(\underline{\boldsymbol{s}}|\boldsymbol{z}, \boldsymbol{A}, \underline{\boldsymbol{\theta}}, \underline{\boldsymbol{x}})$:

$$p(\underline{\boldsymbol{s}}|\boldsymbol{z}, \underline{\boldsymbol{\theta}}, \underline{\boldsymbol{x}}) = \prod_{\boldsymbol{r}} p(\boldsymbol{s}(\boldsymbol{r})|z(\boldsymbol{r}), \boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{r})) = \prod_{\boldsymbol{r}} \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{r}), \boldsymbol{B}(\boldsymbol{r}))$$

with

$$\left\{ \begin{array}{l} \boldsymbol{B}(\boldsymbol{r}) = \left[\boldsymbol{A}^t \boldsymbol{\Sigma}_\epsilon^{-1} \boldsymbol{A} + \boldsymbol{\Sigma}_{z(\boldsymbol{r})}^{-1}\right]^{-1} \\ \boldsymbol{\mu}(\boldsymbol{r}) = \boldsymbol{B}(\boldsymbol{r})[\boldsymbol{A}^t \boldsymbol{\Sigma}_\epsilon^{-1} \boldsymbol{x}(\boldsymbol{r}) + \boldsymbol{\Sigma}_{z(\boldsymbol{r})}^{-1} \boldsymbol{m}_{z(\boldsymbol{r})}] \end{array} \right.$$

$\boldsymbol{s}(\boldsymbol{r})$ is then estimated by its *a posteriori* mean so $\bar{\boldsymbol{s}}(\boldsymbol{r}) = \boldsymbol{\mu}(\boldsymbol{r})$

- $p(\boldsymbol{z}|\boldsymbol{A}, \underline{\boldsymbol{\theta}}, \underline{\boldsymbol{x}})$:

$$p(\boldsymbol{z}|\boldsymbol{A}, \underline{\boldsymbol{\theta}}, \underline{\boldsymbol{x}}) \propto p(\underline{\boldsymbol{x}}|\boldsymbol{z}, \boldsymbol{A}, \underline{\boldsymbol{\theta}})\ p(\boldsymbol{z})$$

where

$$p(\underline{\boldsymbol{x}}|\underline{\boldsymbol{z}}, \boldsymbol{A}, \underline{\boldsymbol{\theta}}) \propto \prod_{\boldsymbol{r}} p(\boldsymbol{x}(\boldsymbol{r})|z(\boldsymbol{r}), \boldsymbol{A}, \underline{\boldsymbol{\theta}}) \propto \prod_{\boldsymbol{r}} \mathcal{N}(\boldsymbol{A}\boldsymbol{m}_{z(\boldsymbol{r})}, \boldsymbol{A}\boldsymbol{\Sigma}_{z(\boldsymbol{r})}\boldsymbol{A}^t + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$$

and

- $p(\boldsymbol{A}|\underline{\boldsymbol{\theta}}, \underline{\boldsymbol{s}}, \boldsymbol{z}, \underline{\boldsymbol{x}})$:

$$p(\boldsymbol{A}|\underline{\boldsymbol{\theta}}, \underline{\boldsymbol{s}}, \boldsymbol{z}, \underline{\boldsymbol{x}}) \propto p(\underline{\boldsymbol{x}}|\boldsymbol{A}, \underline{\boldsymbol{s}}, \boldsymbol{z}, \underline{\boldsymbol{\theta}})\ p(\boldsymbol{A}) \quad \text{with} \quad p(\boldsymbol{A}) = \mathcal{N}(0, \Gamma_p)$$

A is estimated by its maximum *a posteriori*

$$\widehat{\boldsymbol{A}} = \arg\max_{\boldsymbol{A}} p(\boldsymbol{A}|\underline{\boldsymbol{\theta}}, \underline{\boldsymbol{s}}, \boldsymbol{z}, \underline{\boldsymbol{x}})$$

$$\widehat{\boldsymbol{A}} = \left[\sum_{\boldsymbol{r}} \boldsymbol{x}(\boldsymbol{r})\bar{\boldsymbol{s}}'(\boldsymbol{r})\right] \left[\sum_{\boldsymbol{r}} \bar{\boldsymbol{s}}(\boldsymbol{r})\bar{\boldsymbol{s}}'(\boldsymbol{r}) + \boldsymbol{B}(\boldsymbol{r}) + \Gamma_p\right]^{-1}$$

5 Simulation results

For the evaluation of the results, we consider two cases: In simulation data [3], sources $\boldsymbol{s}(\boldsymbol{r})$, the common segmentation $z(\boldsymbol{r})$ and the mixing matrix $\boldsymbol{A}$ are available. So, we can compare the estimated sources $\widehat{\boldsymbol{s}}(\boldsymbol{r})$ with $\boldsymbol{s}(\boldsymbol{r})$, the estimated mixting matrix $\widehat{\boldsymbol{A}}$ with $\boldsymbol{A}$ and the estimated $\widehat{z}(\boldsymbol{r})$ with $z(\boldsymbol{r})$.In the case of real data we compare $\widehat{\boldsymbol{x}}(\boldsymbol{r}) = \widehat{\boldsymbol{A}}\widehat{\boldsymbol{s}}(\boldsymbol{r})$ with $\boldsymbol{x}(\boldsymbol{r})$, For comparing real valued signals $\boldsymbol{s}(\boldsymbol{r})$ and $\boldsymbol{x}(\boldsymbol{r})$, we propose to use the following Lp distances:
$\Delta_p^s = |\boldsymbol{s} - \widehat{\boldsymbol{s}}|^p = \sum_{\boldsymbol{r}} |\boldsymbol{s}(\boldsymbol{r}) - \widehat{\boldsymbol{s}}(\boldsymbol{r})|^p$ and $\Delta_p^x = |\boldsymbol{x} - \widehat{\boldsymbol{x}}|^p = \sum_{\boldsymbol{r}} |\boldsymbol{x}(\boldsymbol{r}) - \widehat{\boldsymbol{x}}(\boldsymbol{r})|^p$ for $p = 1$ and $p = 2$.

For comparing the segmentation results $z(\boldsymbol{r})$, we can use the number of missclassified pixels: $\Delta^z = \sum_{\boldsymbol{r}} \delta(z(\boldsymbol{r}) - \widehat{z}(\boldsymbol{r}))$.

When the matrix $\boldsymbol{A}$ is available, then we can also use: $\Delta_p^A = |\boldsymbol{A} - \widehat{\boldsymbol{A}}|^p = \sum_i \sum_j |A_{ij} - \widehat{A}_{ij}|^p$. However, in most ICA methods, a separating matrix $\boldsymbol{B}$ is estimated and not $\boldsymbol{A}$. Then, we use the generalized inverses $\boldsymbol{B}^t(\boldsymbol{B}\boldsymbol{B}^t)^{-1}$ to be compared with $\boldsymbol{A}$.

BSS Method	$\Delta_1^A(sd)$	$\Delta_2^A(sd)$	$\Delta_1^x(sd)$	$\Delta_2^x(sd)$	$\Delta_1^x(rd)$	$\Delta_2^x(rd)$
PCA	0.42	0.24	4, 18	30, 84	$1, 14.10^3$	$3.05.10^3$
ICA	0.45	0.27	0, 36	0, 33	$1, 37.10^3$	$6, 3.10^6$
proposed method	0.23	0.09	0, 04	0, 004	0, 11	0, 03

Table 1: Mixture matrix performances, distances between data and reconstructed data on simulated images (sd) and distances between data and reconstructed data on real images (rd)

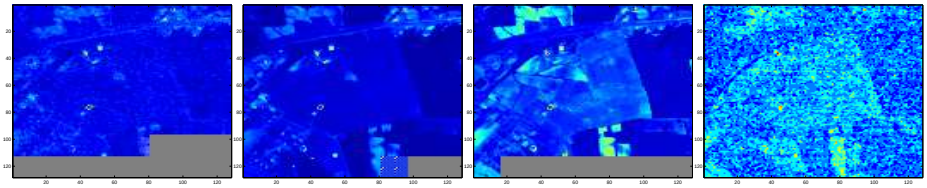

Fig. 2: 4 images extracted from 56 real AVIRIS hyperspectral data: band 1,5,30,40

6 Conclusion

In this paper, we considered the data reduction problem in hyperspectral images as a BSS and presented a Bayesian estimation approach with a particular hierarchical prior model for the observations and sources which gives us the possibility to jointly do data reduction, classification of spectra and segmentation of the images.

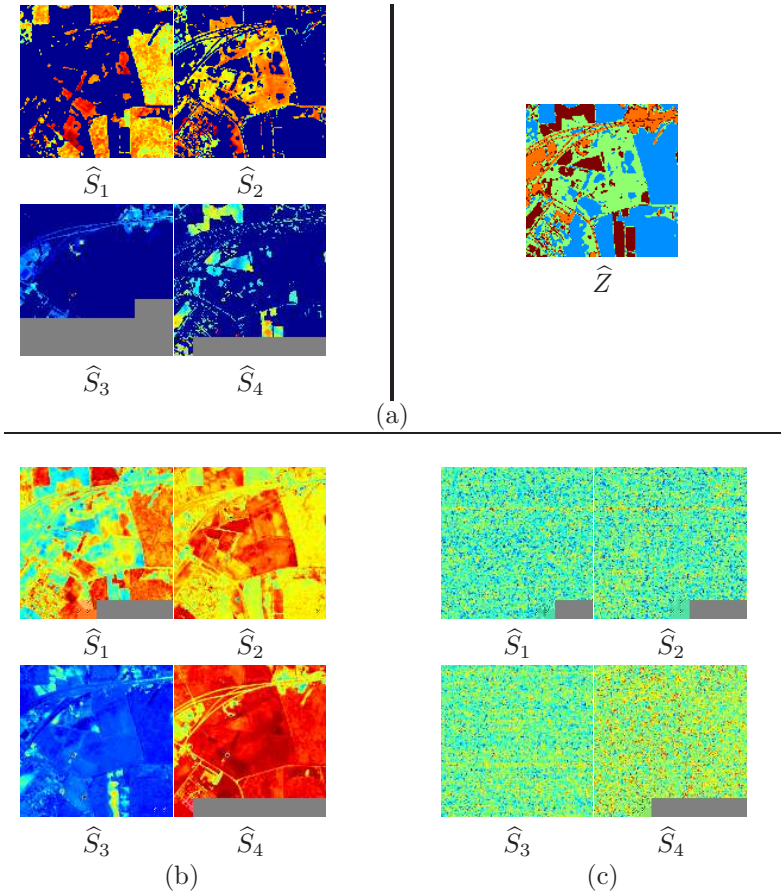

Fig. 3: results obtained size of data (128 × 128)pixel ×56 spectral band (a) proposed method reduction with segmentation (b) ICA (c) PCA. The number of sources is fixed to 4

References

[1] K. Sasaki, S. kawata, and S. Minami, "Component analysis of spatial and spectral patterns in multispectral images," *I. Basics Journal of the Optical Society of America*, vol. 4(11), pp. 2101–2106, 1987.

[2] C. Parra, A. Spence, A. Zieche, K.-R. Mueller, and P. Sajda, "Unmixing hyperspectral data," *In Advances in Neural Information Processing Systems*, vol. 13, pp. 848–854, 2000.

[3] Nadia BALI and Ali Mohammad-Djafari, "Mean field approximation for bss of images with compound hierarchical gauss-markov-potts model," in *MaxEnt05*, august 2005.