Markov Regularization of Mixture of Latent variable Models for Multi-component Image Unsupervised Joint Reduction/Segmentation

F. Flitti and Ch. Collet
LSIIT UMR CNRS-ULP 7005
Université Strasbourg I (ULP)
Bd S. Brant, BP 10413, F-67413 Illkirch, France.
lastname@lsiit.u-strasbg.fr

Abstract This paper is concerned with Multi-component image segmentation which plays an important role in many imagery applications. Unfortunately, we are faced with the Hughes phenomenon when the number of components increases, and a space dimensionality reduction is often carried out as a preprocessing step before segmentation. An interesting solution is the mixtures of latent variable models which recover clusters in the observation structure and establish a local linear mapping on a reduced dimension space for each cluster. Thus, a globally nonlinear model is obtained to reduce dimensionality. Furthermore, a likelihood to each local model is often available which allows a well formulation of the mixture model and a maximum likelihood based decision for the clustering task. However for D-component images classification, such clustering, based only on the distance between observations in the D-dimensional space is not adapted since it neglects the observation spatial locations in the image. We propose to use a Markov a priori associated with such models to regularize D-dimensional pixel classification. Thus segmentation and reduction are performed simultaneously. In this paper, we focus on the Probabilistic Principal Component Analysis (PPCA) as latent model, and the Hidden Markov quad-Tree (HMT) as a Markov a priori.

Keywords: multi-component image segmentation, dimensionality reduction, mixture of latent variable model, hidden Markov quad-tree, Regularization.

1 Introduction

Multi-component image segmentation is of great interest in many imagery applications like medicine, remote sensing, astronomy and non destructive Control. Segmentation algorithms learn data structure to gather pixels according to a given measure, under some smoothness constraints. Generally, they require sufficient observations to correctly estimate model parameters. For multi-component images the required number of samples grows quickly along with the dimension (i.e., number of components) so that the segmentation accuracy decreases rapidly in practice. This is the curse of dimensionality or Hughes phenomenon [7] which consists of the loss of model estimation accuracy as dimensionality grows. To address this problem, one may carry out a space dimensionality reduction as a preprocessing step [12]. Fortunately, high dimensional observations can often be described in a significantly smaller dimension than the original due to the redundancy in neighboring components. Many approaches were proposed in the last decade. All of them seek a mapping onto a reduced dimension space by maximizing a given criterion [1, 8, 6]. Generally data with complex structures requires non linear mapping [19]. Thus, several works has been proposed to develop non linear models. One attracting way is to use a collection of locally linear models, so that each observation is modeled using either a single local model [9] or a mixture of all local models [19, 13, 3]. In the first case, observations are hardly partitioned into clusters (i.e., classes) each one spanned by a local linear model. However, in the second case each observation generation responsibility is shared by all clusters. When probabilistic models are associated with such local models, the observation likelihood can be computed using a well formulated solution of the mixture model parameter estimation problem [19, 13, 3]. Nevertheless for D-components images, the clustering established by such mixture models is based only on the closeness of the observations (pixels) in the D-dimensional space and do not take into account their proximity in the image. We propose to use a Markov a priori associated with such this model to regularize D-dimensional pixel classification. In this way, the pixels are classified based on their closeness both in the RD space and in the H × W lattice. Thus, segmentation and reduction are performed simultaneously avoiding the Hughes phenomenon. In this paper, we focus on the Probabilistic Principal Component Analysis (PPCA) as latent variable model, and the Hidden Markov quad-Tree (HMT) as a Markovian a priori. However, our method may be extended to other latent variable models like factor analyzers and independent component analyzers and other Markov models such as Hidden Markov Chain and Markov...
Random Field.

The paper is organized as follows. The Hidden Markov quad-Tree (HMT) model is described in section 2. In section 3, the PPCA and the Mixture of Probabilistic Principal Component Analyzers (MP-PCA) are presented and linked to the HMT. Then, in section 4, the advantages of the method are exhibited. Section 5 is reserved to the experiments on synthetic and real multi-component images. Finally, conclusions and possible extensions are presented.

2 Hidden Markov quad-Tree

Hidden Markov models have proved to be robust and efficient image analysis methods. In the context of multi-component images, handling often correlated observed data requires a well-designed modeling framework. Resorting to a Bayesian scheme based on Markov models is indeed attractive when dealing with large amount of multi-component observations. Nevertheless, the well known Markov Random Fields (MRF) lead to iterated optimization algorithms exhibiting a high computation burden [2], even if some strategies to decrease the computing cost have been proposed in the last decade [5, 10, 15, 14]. This is due to the fact that most of Markov models are non-causal. As a consequence, the exact inference solution is not computable but only approximated iteratively, which might turn prohibitively expensive. One way to circumvent this problem is to resort to a Markov model on a quadtree where in-scale causality permits non-iterative inference [11, 17] as in the case of hidden Markov chains [4].

A Hidden Markov Tree (HMT) is an acyclic graph \( G = (S, L) \) with a set of nodes \( S \) and a set of edges \( L \). \( S \) is partitioned into “scales”, i.e., \( S = S^0 \cup S^1 \cup \ldots \cup S^R \), such that \( S^R = \{r\} \) is the root, \( S^n \) involves \( 4^{R-n} \) nodes, and \( S^0 \) is the finest scale formed by the leaves. Each node \( s \), excepts the root \( r \), has a unique predecessor, its “parent” \( s^- \). Each node \( s \), expects the “leaves”, has four “children” \( s^+ = \{u \in S : u^- = s \} \). We note also \( s^{++}, s^{++} \), all the descendent of \( s \).

Let the hidden process \( X \) which assigns to each node \( s \in S \) a hidden state \( X_s \) chosen from the label set \( \Omega = \{\omega_1, \ldots, \omega_K\} \) of the \( K \) classes. \( X \) is assumed Markovian in scale, i.e.,:

\[
P(x^n|x^k, k > n) = P(x^n|x^{n+1}), \quad x^n = \{x_s : s \in S^n\}.
\]

Moreover, \( X_s, s \in S^n \), is independent from all \( X_u, u \in S^{n+1} \), given its parent and the inter-scale transition probability can be factorized in the following way [11]:

\[
P(x^n|x^{n+1}) = \prod_{s \in S^n} P(x_s|x_{s^-})
\]

The hidden process \( X \) is called Markov tree since it verifies [16]:

\[
P(x) = P(x_r) \prod_{n=0}^{R-1} \prod_{s \in S^n} P(x_s|x_{s^-})
\]

\(1\)To simplify notation, we will denote the discrete probability \( P(X = x) \) as \( P(x) \).

The multi-component observations \( Y \) are introduced at the scale \( S^0 \) so that each \( D \)-dimensional pixel \( y_s \) is linked to the hidden state \( X_s \) (Fig. 1). The HMT assumes \( y_s \) independent from all the quad-tree given its hidden state, which is formulated as follows:

\[
P(y_s/x, y - \{y_s\}) = P(y_s/x_s).
\]

Thus the probability of \( Y \) conditionally to \( X \) is expressed as the following product:

\[
P(y|x) = \prod_{s \in S^0} P(y_s|x_s),
\]

where \( \forall s \in S^0, P(y_s|x_s = \omega_i) \), called data driven term, captures the likelihood of the observation \( y_s \) wrt the class \( \omega_i \). If no data is available at a given site \( s \) in the image, i.e., missing or masked data, the likelihood at this site is set to 1. In Sect. 3, we link MPCCA to the HMT so that each class is spanned by a local PPCA model. Thus \( P(y_s|x_s = \omega_i) \) is computed as the likelihood of \( y_s \) wrt the local PPCA modeling of the class \( \omega_i \).

From the assumptions above, the joint distribution \( P(x, y) \) can easily factorized as follows:

\[
P(x, y) = P(x_r) \prod_{s \neq r} P(x_s|x_{s^-}) \prod_{s \in S^0} P(y_s|x_s).
\]

The HMT parameters are:

- \( \Phi_x \) the a priori parameters regrouping:
  - \( \{\pi_i = p(x_r = \omega_i)\}_{i=1, \ldots, K} \) the probabilities at the root,
  - \( \{\alpha_{ij} = p(x_s = \omega_j/x_{s^-} = \omega_i)\}_{i,j=1, \ldots, K} \) the parent/child transition probabilities,

Figure 1: Example of a dependency graph corresponding to a quadtree structure on a 4x4 lattice. White circles represent labels and black circles represent multi-band observations \( y_s, s \in S \). Each node \( s \) has a unique parent \( s^- \), and four children \( s^+ = \{s_1^+, \ldots, s_4^+\} \).
with the following updates [11]:

\[ P(x_s = \omega_i | y_{s+}) = P(x_s = \omega_i) = \frac{P(x_s = \omega_i)P(y_s/x_s = \omega_i)}{\sum_{\omega_j} P(x_s = \omega_j)P(y_s/x_s = \omega_j)}, \]

where \( P(x_s = \omega_i) \) is recursively evaluated through a top-down pass, given the prior probability \( P(x_r = \omega_i) = \pi_i \) as follows:

for \( n = R - 1, \ldots, 0 \) do
  for all \( s \in S^n \) do
    \( P(x_s = \omega_i) = \sum_{\omega_j} P(x_s = \omega_i/x_s = \omega_j)P(x_s = \omega_j) \)
  end for
end for.

- **Upward pass**:
  for \( n = 1, \ldots, R \) do
    for all \( s \in S^n \) do
      \( P(x_s = \omega_i | y_{s+}) = \frac{1}{Z} P(x_s = \omega_i) \Pi_{t \in s} a_{ij}P(x_j = \omega_j/y_{s+})/P(x_j = \omega_j) \)
    end for
end for

where \( a_{ij} = P(x_t = \omega_j/x_s = \omega_i) \) is the parent/child transition probability and \( Z \) is a normalizing factor such that \( \sum_{\omega_i} P(x_s = \omega_i/y_{s+}) = 1 \). Note that at the top of quadtree we obtain \( P(x_r = \omega_i/y) \).

- **Downward pass**:
  for \( n = R - 1, \ldots, 0 \) do
    for all \( s \in S^n \) do
      \( P(x_s = \omega_j, x_s = \omega_i/y) = P(x_s = \omega_j/y)P(x_s = \omega_i/y) \)
      \( P(x_s = \omega_j/y) = \sum_{\omega_i} P(x_s = \omega_j, x_s = \omega_i/y) \)
    end for
end for

- **\( \Phi_y \)** the parameters of the likelihoods \( \{P(.|x_s = \omega_i)\}_{i=1, \ldots, K} \).

One of the interests of this model is the possibility of computing exactly the posterior marginals \( P(x_r/y) \) and \( P(x_s, x_s/y) \) at each node \( s \) in two passes on the quad-tree (Algorithm 1).

The EM algorithm used for the estimation of the prior parameters \( \Phi_x \), leads to an iterative procedure with the followings updates [11]:

\[
\alpha^{(c+1)}_{ij} = \frac{\sum_{s \in S^n, n > 0} p^{(c)}[s] = \omega_i/y}{\sum_{s \in S^n, n > 0} p^{(c)}[s] = \omega_i/y} \]
\[
\pi^{(c+1)}_i = p^{(c)}[x_r = \omega_i/y] \tag{7} \]

where \([c]\) stands for the current iteration and \( p^{(c)}[s] = \omega_i/y \) and \( p^{(c)}[x_s = \omega_j, x_s = \omega_i/y] \) are computed by way of the two passes of Algorithm 1 using the current parameters.

When converged, i.e., the difference between successive updates is small enough or the maximum of number of iteration reached, the Marginal a Posteriori Mode criterion (MPM) is used to obtain the segmentation map:

\[
\forall s \in S^0, \hat{x}_s = \arg\max_{x_s \in \Omega} p(x_s/y) \tag{8} \]

The estimation of likelihoods parameters \( \Phi_y \) in the case of regularized MPPCA is presented in the next section.

3 Regularization of the mixture of Probabilistic Principal Component analyzers (MPPCA)

3.1 Probabilistic Principal Component analysis (PPCA)

The PPCA is a statistical modeling of the well known PCA, introduced by Tipping and Bishop [20]. It is based on a latent variable model which links each \( D \times 1 \) observed vector \( y_s \) to \( q \times 1 \) latent vector \( t_s, q < D \), as follows:

\[
y_s = At_s + \mu + \epsilon \tag{9} \]

where \( A \) is a \( D \times q \) matrix, \( \mu \) the observed data mean and \( \epsilon \) is an isotropic Gaussian noise, i.e., \( N(0, \sigma^2 I) \), \( I \) being the \( D \times D \) identity matrix.

Thus the probability distribution of \( y_s \) given \( t_s \) is:

\[
P(y_s/t_s) = \mathcal{N}(y_s; At_s + \mu, \sigma^2 I). \tag{10} \]
Choosing Gaussian prior for $t_s$, i.e.:

$$p(t_s) = (2\pi)^{-\frac{D}{2}}e^{\exp\left[-\frac{1}{2}t_s^t t_s\right]}$$  \hspace{1cm} (11)$$
the marginal distribution of $y_s$ is

$$P(y_s) = \mathcal{N}(y_s; \mu, C)$$  \hspace{1cm} (12)$$
with $C = \sigma^2 I + AA^t$ is a $D \times D$ matrix \cite{19}.

Bayes rule gives the a posteriori probability of $t_s$ \cite{19}:

$$p(t_s,y_s) = \mathcal{N}(t_s; M^{-1}A^t(y_s - \mu), M^{-1})$$  \hspace{1cm} (13)$$
where $M = \sigma^2 I - A' A$ is a $q \times q$ matrix.

The maximization of the data log-likelihood $L = \sum_{s \in S^0} \ln\{p(y_s)\}$ gives the following parameter estimators \cite{19}:

$$\hat{\mu} = \frac{\sum_{s \in S^0} y_s}{\text{card}(S^0)}$$

$$\hat{\sigma}^2 = \frac{1}{D - q} \sum_{j=q+1}^{D} \lambda_j$$  \hspace{1cm} (14)$$
$$\hat{A} = U_q(\Lambda_q - \sigma^2 I)^{\frac{1}{2}} R.$$

where $\lambda_j$ are the eigenvalues of the data covariance matrix $\Sigma = \frac{1}{\text{card}(S^0)} \sum_{s \in S^0} (y_s - \mu)(y_s - \mu)^t$ given in descending order ($\lambda_1 \geq \cdots \geq \lambda_q$), $\Lambda_q$ is a diagonal matrix of the $q$ largest eigenvalues, $U_q$ the matrix of the corresponding eigenvectors, and $R$ is an arbitrary orthogonal rotation matrix.

The sum $\sum_{j=q+1}^{D} \lambda_j$ represents the squared error of the approximation of $D$-dimensional vector $y_s$ by $t_s$ on the $q$-dimensional space, $q \leq D$. Thus, it can be efficiently used to estimate the local dimension $q$.

### 3.2 Mixture of Probabilistic Principal Component Analyzers (MPPCA)

The main problem with PPCA (or simply PCA) is that it considers the observed data distribution as a multivariate Gaussian (see Eq. 12) which masks all local structures present in the data. Tipping and Bishop \cite{19} introduce the mixture of Probabilistic Principal Component Analyzers (MPPCA) to model complex data structures as a mixture of local PPCA.

For a $K$ component MPPCA, the observations are partitioned into $K$ clusters (i.e. classes) each one spanned by a local PPCA. Given this model, the distribution of the observations is

$$P(y_s) = \sum_{i=1}^{K} \pi_i P(y_s/x_s = w_i)$$  \hspace{1cm} (15)$$
where the local PPCA corresponding to the classe $\omega_i$ is characterized by the mean $\mu_i$, the variance $\sigma_i^2$, the projection matrix $A_i$ and the prior $\pi_i$.

Note that in this formulation the prior is the same for all $s \in S^0$ and thus no information about the neighborhood is taken into account when classifying $y_s$.

The a posteriori responsibility of the component $i$ for generating the vector $y_s$ is given by :

$$R_{si} = P(x_s = w_i/y_s) = \frac{P(y_s/x_s = w_i)\pi_i}{P(y_s)}$$  \hspace{1cm} (16)$$

The EM algorithm is used to iteratively estimate the mixture parameters \cite{19}:

$$\hat{\pi}_i = \frac{1}{\text{card}(S^0)} \sum_{s \in S^0} R_{si}$$  \hspace{1cm} (17)$$

$$\hat{\mu}_i = \frac{\sum_{s \in S^0} R_{si} y_s}{\sum_{s \in S^0} R_{si}}$$  \hspace{1cm} (18)$$

and $\hat{\lambda}$ and $\hat{\sigma}_i^2$ are given, in the same way as Eq. 14, by eigen decomposition of the a posteriori responsibility-weighted covariance matrix:

$$\Sigma_i = \frac{\sum_{s \in S^0} R_{si}(y_s - \hat{\mu}_i)(y_s - \hat{\mu}_i)^t}{\sum_{s \in S^0} R_{si}}$$  \hspace{1cm} (19)$$

The MPPCA is a powerful well-formulated tool to capture data local structures in the D-dimensional space. However, its use in multi-component images segmentation gives no smooth maps since no information about observation location in the image is taken into account in this modeling. In the next subsection we link the MPPCA principle to the HMT model to regularize the segmentation solution.

### 3.3 Regularized Mixture of Probabilistic Principal Component Analyzers

We adapt MPPCA model by imposing a Markov constraints via the quadtree modelling. The observation probability becomes:

$$P(y_s) = \sum_{i=1}^{K} P(x_s = \omega_i) P(y_s/x_s = \omega_i)$$  \hspace{1cm} (20)$$

where $X$ is a Markov tree (Eq. 3) and each class $\omega_i$ is spanned by a local PPCA. In this way the likelihood $P(y_s/x_s = \omega_i)$ is computed as the likelihood of $y_s$ wrt the local PPCA corresponding to the class $\omega_i$. As Eq. 12 suggests, this likelihood is given by:

$$P(y_s/x_s = \omega_i) = \mathcal{N}(y_s; \mu_i, C_i)$$  \hspace{1cm} (21)$$

The matrix $C_i$ is obtained in analog manner to Eq. 14 by eigen-decomposition of the weighted covariance matrix

$$\Sigma_i = \frac{\sum_{s \in S^0} P(x_s = \omega_i/y)(y_s - \hat{\mu}_i)(y_s - \hat{\mu}_i)^t}{\sum_{s \in S^0} P(x_s = \omega_i/y)}$$  \hspace{1cm} (22)$$

where

$$\hat{\mu}_i = \frac{\sum_{s \in S^0} P(x_s = \omega_i/y)y_s}{\sum_{s \in S^0} P(x_s = \omega_i/y)}$$  \hspace{1cm} (23)$$

The estimation of the a priori parameter remains the same as in the classical quadtree Eq. 7. The whole algorithm is given in Algorithm 2.
Algorithm 2 Regularized MPPCA procedure

- Computation of an initial segmentation map, using K-means algorithm for example, and estimation of the means and variances within each class \( \Phi_y^{[0]} \).
- Initialization of the prior parameters \( \Phi_x^{[0]} \):
  \[
  \pi_i = 1/K, \quad i = 1, \cdots, K \\
  a_{ij} = 3/4 \quad \text{and} \quad a_{ij} = 1/4(K-1), \quad i, j = 1, \cdots, K \\
  c = 1
  \]
- repeat
  - Perform a two passes on the quadtree (Algorithm 1) using \( \Phi_x^{[c]}, \Phi_y^{[c]} \).
  - Compute \( \{ \Phi_x^{[c+1]}, \Phi_y^{[c+1]} \} \) : Eq.7, Eq. 23, Eq. 22 and Eq. 14.
  - \( c = c + 1 \)
- until the difference between \( \{ \Phi_x^{[c]}, \Phi_y^{[c]} \} \) and \( \{ \Phi_x^{[c+1]}, \Phi_y^{[c+1]} \} \) is small enough or the maximum number of iterations is reached
- Obtain the segmentation map using Eq. 8.

5 Experiments

Synthetic images

To test our approach, we generated three sets of three correlated images each, with constant correlation ratio \( \rho = 0.8 \). Each image contains two Gaussian classes representing a geometric shape and background as detailed in Tab. 1. Thus we obtain 9 images to segment (Fig. 2 top). The segmentation maps with 4 classes using the maximum likelihood classifier based on MPPCA and regularized MPPCA are shown in the bottom of Fig. 2. It is very easy to note the effect of the Markov regularization (on the right) compared to the maximum likelihood classification (on the left). More than 99% of the pixels are correctly classified.

Table 1: Parameters of the two Gaussian classes. The third band is completely corrupted with the noise.

<table>
<thead>
<tr>
<th>class 1</th>
<th>class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mu_1 )</td>
<td>( \sigma_1 )</td>
</tr>
<tr>
<td>Band 1</td>
<td>7.75</td>
</tr>
<tr>
<td>Band 2</td>
<td>7.75</td>
</tr>
<tr>
<td>Band 3</td>
<td>8</td>
</tr>
</tbody>
</table>

Figure 2: Synthetic images on the top. Each 3 images of the same line are simulated using the parameter of Tab. 1. On the bottom, segmentation map obtained with the MPPCA (left) is very noisy, whereas the map obtained with the proposed technique (right) is well regularized (error 5.39%).
Remote sensing images

We have also tested our algorithm on real 6-wavelength remote sensing images of Strasbourg area. The corresponding wavelengths are: 1011, 1250, 1802, 2254, 1802, 2254 nm. We perform 6-classes segmentation using regularized MPPCA. The results are presented in the bottom of the Fig. 3. We clearly distinguish the different regions present in the scene: water, vegetation and tracks between vegetation fields. The different vegetation areas are separated according to their behavior in the different wavelengths. This result is very satisfactory.

Figure 3: Segmentation of remote sensing images using regularized MPPCA. From the top left to the right of the third line, the corresponding wavelengths are: 1011, 1250, 1802, 2254, 1802, 2254 nm. In the bottom, the segmentation map 6 classes.

Astronomical images

The third set of test images concerns 32 bands in Visible spectrum around the Oxygen OIII ray (489nm 506nm) of the nebula near the Eta Carinae star (Fig. 4). We mainly observe stars (punctual sources) in the begin and end bands (Fig. 4 bands 3 and 26), and a cloud (diffuse emission) due to the nebula in the central bands (Fig. 4 bands 12, 14, 16 and 18). We perform 8-classes segmentation map using the regularized MPPCA algorithm. We easily distinguish the stars and the nebula cloud. The result is very encouraging, and an astrophysical interpretation still requisite. The use of the classical multivariate Gaussian modeling for this image set is not possible since the covariance matrix requires 528 parameters for each class.

6 conclusion

A Markov regularization of the Mixture of Probabilistic Principal Component Analyzers (MPPCA) for multi-component image joint reduction/segmentation is introduced. This method avoid the well-known curse of dimensionality when handling large number of component. The model parameter estimation is performed using the EM algorithm. The results on synthetic and real images are very satisfactory. In comparison with a Maximum likelihood classification based on MPPCA, our classification shows clearly the benefit of the smoothing effect of the Markov regularization. Moreover, the result obtained with the 32 astronomical band shows clearly the reduction aspect of the method.

This method is very rich, and can be easily extended to other mixtures of latent variable models when a statistical modeling is available. Thus, one may the regularized mixture of factor analyzers based on the work of Ghahramani et al. [3], or the regularized mixture independent component analyzers based on the work of Roberts et al. [18].

References

Figure 4: Nebula near the Eta Carinae star: 6 bands (in inverse video) selected from 32 bands observed around the Oxygen OIII ray (489nm - 506nm).
Figure 5: Result of 8-classes segmentation of the 32 bands around the Oxygen OIII ray of the Nebula near the Eta Carinae star.


