HAL open science

Triplet Markov fields for the classification of complex structure data

Juliette Blanchet, Florence Forbes

To cite this version:

Juliette Blanchet, Florence Forbes. Triplet Markov fields for the classification of complex structure data. [Research Report] RR-6356, INRIA. 2007. inria-00168621v2

HAL Id: inria-00168621

https://inria.hal.science/inria-00168621v2

Submitted on 15 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Triplet Markov fields for the classification of complex structure data

Juliette Blanchet and Florence Forbes

N° 6356

Août 2007

Thème COG

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6356--FR+ENG

Triplet Markov fields for the classification of complex structure data

Juliette Blanchet and Florence Forbes

Thème COG — Systèmes cognitifs
Projet Mistis

Rapport de recherche n° 6356 — Aot 2007 — 30 pages

Abstract: We address the issue of classifying complex data. We focus on three main sources of complexity, namely the high dimensionality of the observed data, the dependencies between these observations and the general nature of the noise model underlying their distribution. We investigate the recent *Triplet Markov Fields* and propose new models in this class that can model such data and handle the additional inclusion of a learning step in a consistent way. One advantage of our models is that their estimation can be carried out using state-of-the-art Bayesian clustering techniques. As generative models, they can be seen as an alternative, in the supervised case, to discriminative Conditional Random Fields. Identifiability issues and possible phase transition phenomena underlying the models in the non supervised case, are discussed while the models performance is illustrated on real data exhibiting the mentioned various sources of complexity.

Key-words: Triplet Markov model, Supervised classification, Conditional independence, High dimensional data, Mixture of Gaussian, EM algorithm, Mean field approximation, Identifiability, Phase transition, Texture recognition.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38334 Montbonnot Saint Ismier (France)
Téléphone : +33 4 76 61 52 00 — Télécopie +33 4 76 61 52 52

Champs de Markov Triplets pour la classification de données à structure complexe

Résumé : Nous abordons le problème de la classification de données à structure complexe. Nous nous intéressons à trois sources de complexité, à savoir la grande dimension des données observées, les dépendances entre ces observations et la nature générale du modèle de bruit sous-jacent à leurs distributions. Nous étudions les récents *modèles de Markov Triplets* et proposons de nouveaux modèles de cette famille permettant de modéliser de telles données, ainsi que d'inclure une phase d'apprentissage de manière consistante. Un avantage de nos modèles est que leur estimation peut être effectuée à l'aide de techniques bayésiennes standards. En temps que modèles génératifs, ils peuvent être vus comme une alternative, dans le cas supervisé, aux modèles discriminatifs de champs de Markov conditionnels. Le problème de l'identifiabilité ainsi qu'un phénomène possible de transition de phase sont discutés. La performance de ces modèles est illustrée sur des données réelles présentant les différentes sources de complexité mentionnées précédemment.

Mots-clés : Modèle de Markov Triplet, Classification supervisée, Indépendance Conditionnelle , Données de grande dimension, Mélange de gaussiennes, Algorithme EM, Approximation en champ moyen, Identifiabilité, Transition de phase, Reconnaissance de textures.

1. Introduction

Statistical methods that were once restricted to specialist statisticians, such as multivariate discrimination and classification, are now widely used by individual scientists, engineers, and social scientists, aided by statistical packages. However, these techniques are still restricted by necessary simplifying assumptions, such as precise measurement and independence between observations, and it long ago became clear that in many areas such assumptions can be both influential and misleading. There are several generic sources of complexity in data that require methods beyond the commonly-understood tools in mainstream statistical packages. In this paper, we consider more specifically classification problems in which observations have to be grouped into a finite number of classes. We propose a unified Markovian framework for classifying unlabelled observed data into these classes. We focus on three sources of complexity. We consider data exhibiting (complex) dependence structures, having to do for example with spatial or temporal association, family relationship, and so on. Markov models or more generally hidden Markov models are used to handle dependencies. Observations are associated to sites or items at various locations. These locations can be irregularly spaced. This goes beyond the regular lattice case traditionally used in image analysis and requires some adaptation. A second source of complexity is connected with the measurement process, such as having multiple measuring instruments or computations generating high dimensional data. There are not so many one dimensional distributions for continuous variables that generalize to multidimensional ones except when considering product of one dimensional independent components. The multivariate Gaussian distribution is the most commonly used but it suffers from significant limitations when it comes to modelling real data sets. For very high dimensional data, the general covariance matrix model involves the estimation of too many parameters, leading to intractable computations or singularity issues. Solutions have been proposed based on so-called *parsimonious models* [10, 1] but they are not specifically designed for high dimensional data. They do not take into account the fact that real data points are often confined to a region of the space having lower effective dimensionality, so that the data actually live on a smaller dimensional manifold embedded within the high dimensional space. Other approaches consider reducing the dimensionality of the data as a preprocessing step possibly using Principal Component Analysis or variable selection methods. In a classification context, this may not be satisfactory as relevant information may be lost that can help separating the classes. For these reasons, we rather consider a more recent approach developed for independent Gaussian mixtures [7]. We extend this approach to our Markov random field models and maintain this way their efficiency and tractability for high dimensional data. Both dependencies between sites and dependencies between components of the multidimensional observations are modelled while the number of parameters to be estimated remains tractable. Another limitation of Gaussian distribution is that a single Gaussian distribution is unable to capture non unimodal structures. Also, we consider as a source of complexity the fact that in real-world applications, data cannot usually be reduced to classes modelled by unimodal distributions and consequently by single Gaussian distributions. As regards the measurement process, we propose class and site dependent mixtures of distributions and more specifically mixtures of Gaussian distributions which provide a richer class of density models than the single Gaussian distributions. A third major source of complexity is related to the structure of the *noise* model or the distribution linking the unknown labels to the observations. A strong assumption of conditional independence of the observed data is gen-

erally used in the hidden Markov field framework for tractability. This assumption combined with the Markovianity of the hidden field has the advantage to lead to a distribution of the labels given the observations (the *posterior distribution*) which is Markovian. This last property is essential in all Markov model based clustering algorithms. However, conditional independence is too restrictive for a large number of applications such as textured or non-stationary image segmentation. For this reason, various Markov models have been proposed in the literature including Gaussian Markov fields [12] and more recently *Pairwise Markov models* [28]. The latter are based on the observation that the conditional independence assumption is sufficient but not necessary for the Markovianity of the conditional distribution to hold. A further generalization has then been proposed in [2] through the *Triplets Markov models* with larger modelling capabilities, allowing more general noise models and in particular multi-modal class distributions. In practice, the Triplet models illustrated in applications (see [2] and [3]) satisfy particular assumptions. In this paper, we consider Triplet models different from those in [2] and [3]. Our models were originally designed for supervised classification issues in which training sets are available and correspond to data for which data exemplars have been grouped into classes. Non trivial extensions are required to include a learning step while preserving the Markovian modelling of the dependencies. In this context, we propose a class of Triplet Markov models with rich noise models that still allow standard processing as regards classification and parameter estimation. We illustrate our models using an Expectation Maximization framework and a mean field like approximation procedure developed in [9] for the standard Hidden Markov field case. Any other choice (Iterated Condition Expectation, Stochastic gradient, etc.) would have been possible but it is not the purpose of this paper to provide a comparison of all these techniques. We extend the approach in [9] to the estimation and use of our Triplet models including a learning and test stages in the supervised case. We consider the issue of selecting the best model with regards to the observed data using a criterion based on the Bayesian Information Criterion (BIC). When studying these Triplet Markov models, we also illustrate on a simple example, that the underlying Markov fields can undergo a phase transition phenomenon with possible consequence on their parameter estimation. Although more general, it is important to note that our Triplet models are simpler to deal with and have greater modelling capabilities in a supervised case. In the non supervised case, general noise models can lead to non-identifiability issues. Also, it is important to specify the relationship between the Triplet Models and the Conditional Random Fields (CRF) [21] which have been widely and successfully used in applications including text processing, bioinformatics and computer vision. CRF's are *discriminative models* in the sense that they are based on and they model directly the posterior or conditional distribution of the labels given the observations. This has the advantage not to require the modelling of the distribution of the observed data alone and the conditional independence assumption. Explicit models of the joint distribution of the labels and observations or of the noise distribution are not required. In classification issues, the posterior distribution is the one needed and it can appear as a waste of time and computational ressources to deal with the joint distribution or with complex noise models. As summarize in the following sentence from [32] p.158: "One should solve the [classification] problem directly and never solve a more general problem as an intermediate step [such as the modelling of the joint distribution]". All the more so as the class conditional distributions describing the noise model may contain a lot of structure but with little effect on the posterior distribution (see for instance Figure 1.27 in [5]). However, even in classification contexts,

approaches that model the joint distribution of the labels and observations are considered. They are known as *generative models*. Triplet Markov models belong to this class. Such generative models are certainly more demanding in term of modelling but they have the advantage to provide a model of the observed data (the likelihood) allowing this way better access to theoretical properties of the estimators usually based on a maximum likelihood principle. This can also be useful for detecting outliers or data points that have low probability under the model and for which the predictions may be of low accuracy. Even when the main interest is classification, such outliers if not detected can severely biased the parameter estimates and affect the classification results. There has been growing interest in exploring the relative merits of generative and discriminative approaches and in finding ways to combine them. In more and more modern applications, learning data is not enough and the use of external information including a priori or expert knowledge is necessary. Such expertise is usually embedded in the data structure which can be taken into account by using generative models.

In this work, we show that Triplet Markov models can then be seen as an alternative to Conditional Random Fields with good modelling capabilities. As generative models they better model the structure of the data. They can be used with standard Bayesian techniques and probabilistic clustering tools requiring no more algorithmic effort than CRF's. They allow theoretically well-based studies and in particular model selection to guide the user to specific modelling choices consistent with the observed data.

To outline the structure of the paper, the hidden Markov model approach is recalled in the following section which presents basic tools and points out some limitations when dealing with complex data. Section 3 introduces our *Triplet Markov Field* model in the context of supervised segmentation. A general scheme and procedure based on EM like algorithms for parameter estimation is proposed in Section 4. The automatic selection of Triplet Markov models is addressed in Section 5. As an illustration, simulations of a simple Triplet model are shown in Section 6 and a phase transition phenomenon for the underlying Markov field is pointed out. In Section 7, we consider a texture recognition task which involves real complex data. Section 8 ends the paper with elements for discussion and further work.

2 Hidden Markov Model based clustering

Hidden structure models and more specifically Gaussian mixture models are among the most statistically mature methods for clustering. The success of hidden structure models lies partly in the fact that clustering can be seen as a labelling problem and therefore corresponds to many problems in practice. A labelling problem is specified in terms of a set of sites S and a set of labels $\mathcal{L}$. A site often represents an item, a point or a region in the Euclidean space such as an image pixel or an image feature such as a corner, a line segment or a surface patch. A set of sites may be categorized in terms of their regularity. Sites on a lattice are considered as spatially regular (*eg.* the pixels of a 2D image). Sites which do not present spatial regularity are considered as irregular. This is the usual case when sites represent geographic locations or features extracted from images at a more abstract level, such as the detected corners and lines and more generally *interest points* (see Section 7). It can also be that the sites correspond to items (*eg.* genes) that are related to each other through a distance or dissimilarity measure [33] or simply to a collection of independent items. A label is an event

that may happen to a site. We will consider only discrete label sets. In this case, a label assumes a discrete value in a set of L labels. In edge detection, for example, the label set is the two component set $\{edge, non-edge\}$ [16]. In the following developments, it is convenient to consider $\mathcal{L}$ as the set of L-dimensional indicator vectors $\mathcal{L} = \{e_1, \ldots, e_L\}$ where each e_l has all its components being 0 except the lth which is 1. The labelling problem is to assign a label from a label set $\mathcal{L}$ to each of the sites. If there are n sites, the set $\mathbf{y} = \{y_1, \ldots, y_n\}$ with $y_i \in \mathcal{L}$ for all $i \in S$, is called a labelling of the sites in S in terms of the labels in $\mathcal{L}$. When each site is assigned a unique label, a labelling can be regarded as a function with domain S and image $\mathcal{L}$. In mathematical programming a labelling is also called a coloring, in the terminology of random fields it is called a configuration. In vision, it can corresponds to an edge map, an interpretation of image features in terms of object features, or a pose transformation and so on.

Our approach of the labelling problem aims at modeling dependencies or taking into account contextual information. It is based on hidden Markov models. We consider cases where the data naturally divide into observed data $\mathbf{x} = \{x_1, \ldots, x_n\}$ and unobserved or missing data $\mathbf{y} = \{y_1, \ldots, y_n\}$. The missing data y_i represents the memberships to one of a set of L alternative categories, *ie.* the labels. They are considered as random variables denoted by $\mathbf{Y} = \{Y_1, \ldots, Y_n\}$. When the Y_i's are independent, the model reduce to a standard mixture model. When the Y_i's are not independent, the inter-relationship between sites can be maintained by a so-called neighborhood system usually defined through a graph. Two neigboring sites correspond to two nodes of the graph linked by an edge. The dependencies between neighboring Y_i are then modelled by further assuming that the joint distribution of $Y_1, \ldots, Y_n$ is a discrete Markov Random Field (MRF) on this specific graph defined by

$$P(\mathbf{y}) \quad = \quad W^{-1} \exp(-H(\mathbf{y})) \tag{1}$$

where W is a normalizing constant and H is a function assumed to be of the following form (we restrict to pair-wise interactions), $H(\mathbf{y}) = \sum_{i \sim j} V_{ij}(y_i, y_j)$, where the V_{ij}'s are functions referred to as pair potentials. We write $i \sim j$ when sites i and j are neighbors on the graph, so that the sum above is only over neighboring sites. We consider pair potentials V_{ij} that depend on y_i and y_j but also possibly on i and j. Since the y_i's can only take a finite number of values, for each i and j, we can define a $L \times L$ matrix $\mathbb{V}_{ij} = (\mathbb{V}_{ij}(k,l))_{1 \leq k,l \leq L}$ and write without lost of generality $V_{ij}(y_i, y_j) = -\mathbb{V}_{ij}(k,l)$ if $y_i = e_k$ and $y_j = e_l$. Using the indicator vector notation and denoting y_i^t the transpose of vector y_i, it is equivalent to write $V_{ij}(y_i, y_j) = -y_i^t \, \mathbb{V}_{ij} y_j$. This latter notation has the advantage to still make sense when the vectors are arbitrary and not necessarily indicators. This will be useful when describing the algorithms of Section 8.1. If for all i and j, $\mathbb{V}_{ij} = \beta \times I_L$ where β is a scalar and I_L is the $L \times L$ identity matrix, the model parameters reduce to a single scalar interaction parameter β and we get the Potts model traditionally used for image segmentation [4]. Note that this model is most of the time appropriate for classification since, for positive β, it tends to favor neighbors that are in the same class. However, cases where the $\mathbb{V}_{ij}$'s are far from $\beta \times I_L$ could be useful in situations where neighboring sites are likely to be in different classes. In practice these parameters can be tuned according to expert or *a priori* knowledge or they can be estimated from the data. In the latter case, the part to be estimated is usually assumed independent of the indices i and j, so that in what follows the Markov model parameters will reduce to a single matrix $\mathbb{V}$. Note that

formulated as such the model is not identifiable in the sense that different values of the parameters, namely $\mathbb{V}$ and $\mathbb{V} + \alpha\mathbb{1}$ (where $\mathbb{1}$ denotes the $L \times L$ matrix with all its components being 1) lead to the same probability distribution. This issue is generally easily handled by imposing some additional constraint such as $\mathbb{V}(k,l) = 0$ for one of the components (k,l). A standard example is that of the Potts model for which $\mathbb{V}(k,l) = 0$ for all $k \neq l$.

Hidden Markov Fields (HMF) have been widely used for a number of classification tasks. Most applications are related to image analysis [26] but other examples include population genetics [15], bioinformatics [33], etc.. Standard models are based on the following usual assumptions. Let $\mathbf{X} = (X_i)_{i \in S}$ and $\mathbf{Y} = (Y_i)_{i \in S}$ be two random fields, each X_i taking its values in $\mathbb{R}^d$ and each Y_i taking its values in $\mathcal{L} = \{e_1, \ldots, e_L\}$. $\mathbf{Y}$ is not observed and considered as hidden. In most existing approaches, the hidden field $\mathbf{Y}$ is assumed to be Markovian (equation (1)) with respect to a neighboring system. This is not strictly necessary (see for instance [2]). In a segmentation or classification context, it has the advantage to provide some insight and control on the segmentation regularity through a meaningful and easy to understand parametric model but it also somewhat reduces the modeling capabilities of the approach. In the following developments, we will consider more general cases. The goal is to estimate $\mathbf{Y}$ from the observed $\mathbf{X} = \mathbf{x}$. Most approaches then fall into two categories. The first ones focus on finding the best $\mathbf{y}$ using a Bayesian decision principle such as Maximum A Posteriori (MAP) or Maximum Posterior Mode (MPM) rules. This explicitly involves the use of $P(\mathbf{y}|\mathbf{x})$ and uses the fact that the conditional field denoted by $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ is a Markov field. This includes methods such as ICM [4] and Simulated Annealing [16] which differ in the way they deal with the intractable $P(\mathbf{y}|\mathbf{x})$ and use its Markovianity. A second type of approaches is related to a missing data point of view. Originally, the focus is on estimating parameters when some of the data are missing (the y_i's here). The reference algorithm in such cases is the Expectation-Maximization (EM) algorithm [13]. In addition to providing estimates of the parameters, the EM algorithm provides also a classification $\mathbf{y}$ by offering the possibility to restore the missing data. However, when applied to hidden Markov fields, the algorithm is not tractable and requires approximations. This approach includes among others the Gibbsian EM of [11], the MCEM algorithm and a generalization of it [29], the PPL-EM algorithm of [29] and various Mean Field like approximations of EM [9]. Such approximations are also all based on the Markovianity of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$. This property appears as a critical requirement for any further developments. To our knowledge, there are no approaches to Markov model based clustering that can be carried out without this assumption.

When $\mathbf{Y}$ is Markovian, a simple way to guarantee the Markovianity of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ is to further assume that

$$P(\mathbf{x}|\mathbf{y}) = \prod_{i \in S} P(x_i|y_i) \ . \tag{2}$$

Indeed, equations (1) and (2) imply that $(\mathbf{X}, \mathbf{Y})$ is a Markov Random Field which implies that $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ is a MRF too. This standard and widely used case is referred to, in [2], as the HMF-IN model for Hidden Markov Field with Independent Noise. Equation (2) is a conditional independence and non correlated noise condition. In addition, in such a setting, the class dependent distribution $P(.|y_i)$ is usually a standard distribution, typically a Gaussian distribution $\mathcal{N}(.|\theta_{y_i})$, where the y_i subscript in θ_{y_i} indicates that the distribution parameters depends on the specific value of y_i. More generally, HMF-IN parameters are denoted by $\mathbf{\Psi} = (\Theta, \mathbb{V})$ with $\Theta = \{\theta_1, \ldots, \theta_L\}$. In the one

dimensional Gaussian case, $\theta_{y_i} = (\mu_{y_i}, \sigma^2_{y_i})$ the mean and variance parameters. This corresponds to the following equation:

$$P(\mathbf{X} = \mathbf{x}|\mathbf{Y} = \mathbf{y}) \propto \exp(-1/2 \sum_{i \in S} \sigma^{-2}_{y_i}(x_i - \mu_{y_i})^2). \tag{3}$$

However, the Gaussian assumption is not satisfactory whenever the goal is to partition data into non-homogeneous class for which the distribution of individuals in the class is very unlikely to be Gaussian and more generally unimodal. As an example, this is typically a problem, when trying to segment images into different textures. The last two assumptions (2) and (3) are too simple to allow one to take texture into account. In particular for *complex* classes, it may be critical to capture spatial noise correlations that potentially contain some useful information. For texture modeling, an alternative hypothesis is that textures are realizations of a Gaussian Markov random field [12]. For illustration, in the one dimensional case, it corresponds to:

$$P(\mathbf{x}|\mathbf{y}) \propto \exp(-\sum_{i \sim j} \alpha_{y_i y_j} x_i x_j - \sum_{i \in S} (\alpha_{y_i y_i} x_i^2 + \gamma_{y_i} x_i)). \tag{4}$$

Note the additional double terms $\alpha_{y_i y_j} x_i x_j$ when comparing to equation (3). If the cardinality of S is n, the later corresponds to a multidimensional Gaussian distribution with a n-dimensional diagonal covariance matrix while (4) corresponds to a more general covariance matrix. When defined by (4) the X_i's are not conditionally independent given $\mathbf{Y}$ but the trouble with (4) is that except in particular cases, neither $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ nor $(\mathbf{X}, \mathbf{Y})$ is Markovian. Note that if $(\mathbf{X}, \mathbf{Y})$ is a MRF then $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ is a MRF too but the reverse is not necessarily true. Different strategies can then arise. It appears that a lot of theoretical and computational tools have been developed for a Bayesian treatment (MAP or MPM), so that there are significant advantages both theoretically and practically in adapting new models to this framework. The Triplet Markov Field (TMF) of [2] were designed for this purpose. In what follows, we build new models that are appropriate for Bayesian supervised segmentation of complex data. They can be seen as particular TMF. In particular, complex dependencies in the observed data as well as in the labels can be modeled in a principled manner. The generaly used strong assumption of conditional independence of the observed data can be relaxed. We consider a generative framework but some aspects of this work are similar to the Conditional Random Fields (CRF) approach which considers a discriminative framework and models the conditional distribution $P(\mathbf{y}|\mathbf{x})$ directly [21]. Such CRF s are related to the Pairwise Markov random fields (PMF) of [28]. PMF s consist in modelling the joint distibution $P(\mathbf{x}, \mathbf{y})$ as a MRF, which implies that $P(\mathbf{y}|\mathbf{x})$ is a MRF too without modelling explicitly the likelihood $P(\mathbf{x}|\mathbf{y})$ or assuming that $P(\mathbf{y})$ is Markovian. The TMF approach is based on the introduction of a third field $\mathbf{Z}$ so that $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ is Markovian and therefore $P(\mathbf{z}, \mathbf{y}|\mathbf{x})$ is a MRF as a consequence while $P(\mathbf{y}|\mathbf{x})$ is not necessarily one, generalizing this way the Conditional random field approach. More details are given in the following section. We then show in Section 4, show how we can use algorithms developed for HMF-IN for inference in these more general models.

3 Designing Triplet Markov Fields (TMF) for supervised segmentation of complex data

3.1 Supervised segmentation

We first focus on data that exhibit some complexity due to the non unimodality or non Gaussianity of their class dependent distributions. Doing so, we will also propose models that handle high dimensional data as specified in Section 3.3.

When starting from the standard HMF-IN models (Section 2) to cluster data, a natural idea to extend the modeling capabilities of this approach is to decomposed each class, given by the y_i's, into subclasses allowing this way more general class dependent distributions. However introducing such subclasses in a mathematically consistent way is not straightforward. Let us first assume that each of the L classes is decomposed into K sub-classes so that we can introduce additional variables $\{Z_1, \ldots, Z_n\}$ indicating the sub-classes and then consider class and sub-class dependent distributions $P(.|y_i, z_i)$ that depend on some parameters $\theta_{y_i z_i}$. The $\theta_{y_i z_i}$'s belong to a set $\Theta = \{\theta_{lk}, l = 1 \ldots L, k = 1 \ldots K\}$. When considering standard independent mixture models, for instance, introducing an extra subclass variable leads to mixtures of mixtures and then raises some identifiability issues. A mixture of mixtures is simply a larger mixture but a mixture of (*eg.* Gaussian) mixtures is not in general identifiable [18]. Intuitively, at the θ_{lk}'s level, there is some ambiguity when trying to assign each component (subclass) to its class. This lack of identifiability corresponds to a problem known as the *label switching* problem in a Bayesian framework [8]. It is due to the fact that mixtures of components belonging to the same parametric family are invariant under permutation of the component labels. However, when the component parameters $\Theta = \{\theta_{lk}, l = 1 \ldots L, k = 1 \ldots K\}$ are known this identifiability issue disappears. This case occurs in a supervised framework where learning data are used to first estimate the θ_{lk}'s. However how to include a learning step when dealing with Markovian dependent data is not straightforward. In a learning stage typically, observations $\mathbf{x}$ are given together with their ground truth $\mathbf{y}$. This means that both $\mathbf{x}$ and $\mathbf{y}$ are known and that the model parameters $\mathbf{\Psi} = (\Theta, \mathbb{V})$ has to be estimated by maximizing the joint distribution, $P(\mathbf{x}, \mathbf{y}|\mathbf{\Psi})$ over $\mathbf{\Psi}$. It is easy to see that estimating parameters Θ is done independently of the assumptions made on $P(\mathbf{y})$. In particular if (2) holds, whatever the condition on $P(\mathbf{y})$ (Markovianity, etc.), the parameters will be estimated as if the sites i were independent. Relaxing assumption (2) is therefore essential when considering a supervised framework since assuming (2) does not allow to take site dependencies into account. For textures, this has been used in [23] but the approach based on independent Gaussian mixtures and used for texture recognition, cannot be extended in a consistent way. If the class dependent distributions are assumed to be standard mixture of Gaussians, the model used in learning each class is an independent mixture model and does not account for dependencies between the sites. We could deal with learning data as independent mixtures but this will mean dealing with two different models, one for the images in the learning set and one for the images in the test set. As an alternative, in the following, we consider less straightforward but consistent extensions of HMF-IN.

We propose to define the distribution $P(\mathbf{x}|\mathbf{y})$ in a more general way. Equation (3) defines the distribution of $\mathbf{X}|\mathbf{Y} = \mathbf{y}$ as a n-dimensional Gaussian distribution with diagonal covariance matrix

due to the conditional independence assumption. We generalize (3) by introducing an additional field $\mathbf{Z} = (Z_i)_{i \in S}$ with $Z_i \in \mathcal{K} = \{e'_1, \ldots, e'_K\}$ where the e'_k are K-dimensional indicator vectors. For all $\mathbf{y} \in \mathcal{L}^n$, we can write

$$P(\mathbf{x}|\mathbf{y}) = \sum_{\mathbf{z} \in \mathcal{K}^n} P(\mathbf{z}|\mathbf{y}) P(\mathbf{x}|\mathbf{y}, \mathbf{z}) = \sum_{\mathbf{z} \in \mathcal{K}^n} \Pi_{\mathbf{yz}} \, f_{\theta_{\mathbf{yz}}}(\mathbf{x}). \tag{5}$$

The distribution of $\mathbf{X}|\mathbf{Y} = \mathbf{y}$ can be seen as a mixture of K^n distributions where the mixing proportions, denoted by $\Pi_{\mathbf{yz}}$, are the $P(\mathbf{z}|\mathbf{y})$'s and the mixed distributions are denoted by $f_{\theta_{\mathbf{yz}}}(\mathbf{x}) = P(\mathbf{x}|\mathbf{y}, \mathbf{z})$. More specifically, we will consider Gaussian $P(\mathbf{x}|\mathbf{y}, \mathbf{z})$ with independence between the components, *ie.*

$$f_{\theta_{\mathbf{yz}}}(\mathbf{x}) = P(\mathbf{x}|\mathbf{y}, \mathbf{z}) = \prod_{i \in S} f_{\theta_{y_i z_i}}(x_i) = \prod_{i \in S} P(x_i|y_i, z_i), \tag{6}$$

where $\{f_{\theta_{lk}}, l \in \{1, \ldots, L\}, k \in \{1, \ldots, K\}\}$ are d-dimensional Gaussian distributions with parameters $\theta_{lk} = (\mu_{lk}, \Sigma_{lk})$. In particular it follows that for all $i \in S$,

$$P(x_i|y_i) = \sum_{z_i \in \mathcal{K}} P(z_i|y_i) f_{\theta_{y_i z_i}}(x_i) \ ,$$

which is a mixture of K Gaussians depending on y_i and whose mixture coefficients $P(z_i|y_i)$ also depend on the site i. Equation (9) below shows that this latter dependency is one of the key and main differences with standard independent mixtures of Gaussians.

As we do not assume a specific Markovian form for $\mathbf{Y}$, in order to consistently define the full model, *ie.* the joint distribution of $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$, we need to define $P(\mathbf{z}, \mathbf{y})$. We choose a Markovian distribution:

$$P(\mathbf{z}, \mathbf{y}) \propto \exp(\sum_{i \sim j} V_{ij}(z_i, y_i, z_j, y_j), \tag{7}$$

where the $V_{ij}(z_i, y_i, z_j, y_j)$ are pair potentials. These potentials could be written in terms of a $KL \times KL$ matrix $\mathbb{V}$ as specified in Section 2 but we rather write it as

$$V_{ij}(z_i, y_i, z_j, y_j) = z_i^t \mathbb{B}_{y_i y_j} z_j + y_i^t \mathbb{C} y_j \tag{8}$$

where $\{\mathbb{B}_{ll'}, \, l, l' \in \{1, \ldots, L\}\}$ are symmetric matrices of size $K \times K$ so that $\mathbb{B}_{ll'} = \mathbb{B}_{l'l}$ (there is thus $L(L+1)/2$ different matrices) and $\mathbb{C}$ is an additional symmetric matrix of size $L \times L$ that does not depend on the z_i's. This is simply looking at $KL \times KL$ matrix $\mathbb{V}$ as a $L \times L$ matrix of $K \times K$ bloc matrices. The reason for such a parameterization is made clearer below.

It follows from (6) and (7) that the variables $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ are Markovian and consist then in a Triplet Markov Field (TMF) as defined in [2]. From (6) and (7) it comes clearly that as $\mathbf{U} = (\mathbf{Y}, \mathbf{Z})$ is Markovian, the pair $(\mathbf{X}, \mathbf{U})$ is a Gaussian HMF-IN with KL hidden classes. EM like algorithms (*eg.* [9]) can then be applied in particular to provide estimates of the Θ_{lk}'s. However defined as such, the model still suffers from some identifiability issue due to the possibility of label switching. In our case, the aim is to estimate $\mathbf{y}$ from the observed $\mathbf{x}$ using the posterior probability $P(\mathbf{y}|\mathbf{x})$. When

considering the Triplet $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ defined above, this probability is not directly available but only through the marginalization (sum over the $\mathbf{z}$'s) of $P(\mathbf{y}, \mathbf{z}|\mathbf{x})$. In practice, to compute $P(\mathbf{y}|\mathbf{x})$ then requires to sum over the right terms *ie.* to know the permutation of the estimates of the $\{\theta_{lk}, l = 1 \dots L, k = 1 \dots K\}$. This interchanging of labels is generally handled by the imposition of an appropriate constraint on the parameters but none of the usual ones would be general enough and make sense in our context. Other proposals can be found in [25] in a clustering context. They are based on the intuition that components in the same cluster ought to be relatively close to each other which is not true in general (*eg.* texture model). Possibly relabelling techniques using a likelihood or Loss function criterion, as proposed in [31], could be considered but this would required to enumerate to the order of $(KL)!$ permutations at each iteration and will be time consuming even for not so large values of K and L.

The TMF defined above are then not adapted to an unsupervised framework, at least when considering components $f_{\theta_{lk}}$'s from the same parametric family which is often the case when no additional a priori knowledge is available. In a supervised framework this issue disappears, as soon as the $\{\theta_{lk}\}$'s can be learned in a way that allows to group them according to values of k, $\{\theta_{lk}, l = 1 \dots L\}$. The TMF above are appropriate for learning. It follows from (7) that $P(\mathbf{z}|\mathbf{y})$ is Markovian too

$$\Pi_{\mathbf{yz}} = P(\mathbf{z}|\mathbf{y}) = \frac{1}{W(\mathbf{y})} \exp(\sum_{i \sim j} z_i^t \mathbb{B}_{y_i y_j} z_j), \tag{9}$$

where $W(\mathbf{y})$ is a normalizing constant that depends on $\mathbf{y}$. Note that matrix $\mathbb{C}$ disappears in (9). This will result in some variations between the learning and classification steps of Section 4.

Equation (6) means that the X_i's are conditionally independent given the Y_i's and the Z_i's. In the whole model definition it acts in a similar way as equation (2). The keypoint in introducing $\mathbf{Z}$ this way is that given (6) and (9), $\mathbf{X}, \mathbf{Z}|\mathbf{Y} = \mathbf{y}$ is an HMF-IN. This property will be useful in the learning stage while the fact that the pair $(\mathbf{X}, \mathbf{U})$ with $\mathbf{U} = (\mathbf{Y}, \mathbf{Z})$ is an HMF-IN will be useful in the classification stage. More specifically, combining (6) and (9), it comes

$$P(\mathbf{x}, \mathbf{z}|\mathbf{y}) = P(\mathbf{x}|\mathbf{y}, \mathbf{z}) P(\mathbf{z}|\mathbf{y}) = \frac{1}{W(\mathbf{y})} \exp(\sum_{i \sim j} z_i^t \mathbb{B}_{y_i y_j} z_j + \sum_{i \in S} \log f_{\theta_{y_i z_i}}(x_i)), \tag{10}$$

which shows that $P(\mathbf{x}, \mathbf{z}|\mathbf{y})$ does not generally factorize and results then in a different model than the TMF's that [2] (sect 2.5 p.483) and [3] suggest to use in practical applications. The estimation procedures suggested in [2] cannot then be applied straightforwardly but we will propose one in Section 8.1.

3.2 Varying number of sub-classes

As mentioned before, the triplet model is described above for $Z_i \in \mathcal{K}$, meaning implicitly that the number of sub-classes is K for each of the L classes. In practice, it is important to handle the fact that the class distributions may be of various form and in particular the number of sub-classes required to described them may not be the same. To handle this case requires to specify some modifications but does not fundamentally change the procedure.

Let $K_1, \ldots, K_L$ denote the respective desired numbers of sub-classes for classes e_1 to e_L. Let ξ be the set $\xi = \{(e'_k, e_l), l = 1 \ldots L, k = 1 \ldots K_l\}$. The set ξ is included in set $\mathcal{L} \times \{1, \ldots, \max_l K_l\}$.

Keeping the same definition (8) for the potentials, it is enough to replace equation (7) by defining $P(\mathbf{z}, \mathbf{y})$ as follows:

$$P(\mathbf{z}, \mathbf{y}) \propto \exp(\sum_{i \sim j} V_{ij}(z_i, y_i, z_j, y_j), \quad \text{if } (\mathbf{z}, \mathbf{y}) \in \xi \tag{11}$$

$$= 0 \quad \text{otherwise} \tag{12}$$

The factorization property remains and then the various useful Markov properties follow (see *eg.* [22]).

3.3 High dimensional data

Using Gaussian distributions for the $f_{\theta_{lk}}$'s in equation (6) has the advantage to admit a straightforward formulation of the model for high dimensional data. However, estimating full covariances matrices is not always possible and advisable beyond small dimensions. A common solution is to consider diagonal covariance matrices but this is assuming independence between the observations components and is usually not satisfying. As an alternative, we propose to use specific parameterizations described in [7]. The authors propose new Gaussian models of high dimensional data for clustering purposes based on the idea that high dimensional data live around subspaces with a dimension lower than the one of the original subspace. Low dimensional class-specific subspaces are introduced in order to limit the number of parameters. The covariance matrix Σ_{lk} of each class is re-parameterized in its eigenspaces. Denoting by Q_{lk} the orthogonal matrix with the eigenvectors of Σ_{lk} as columns, the class conditional covariance matrix D_{lk} is therefore defined in the eigenspace of Σ_{lk} by $D_{lk} = Q_{lk}^t \Sigma_{lk} Q_{lk}$. The matrix D_{lk} is a diagonal matrix which contains the eigenvalues of Σ_{lk}. It is further assumed that the diagonal of D_{lk} is made of d_{lk} first values, $a_{lk}^1, \ldots, a_{lk}^{d_{lk}}$, and $d - d_{lk}$ other values all fixed to some value b_{lk} with, for all $j = 1, \ldots, d_{lk}$, $a_{lk}^j > b_{lk}$. Notation d denotes the dimension of the original space and $d_{lk} \in \{1, \ldots, p-1\}$ is unknown. See [7] for additional details and further interpretation of such decompositions. In the present work, we recast this approach into the EM based procedure described in Section 8.1. When dealing with high dimensional data, this reduces the number of parameters to be estimated significantly and tends to avoid numerical problems with singular matrices while allowing to go beyond the standard diagonal covariance matrices and the usual independence assumptions between dimensions.

4 The supervised classification of complex data scheme

More than an algorithm, we describe a general scheme to deal with complex data as specified. As regards parameter estimation, we consider a soft clustering approach and use an algorithm based on EM and *mean field*-like approximations [9]. The algorithm is decribed in Appendix. We implemented it to illustrate the performance of the models we propose but other algorithms could be considered and would fit the scheme. Its actual use in our supervised classification framework

requires two stages which are described in Sections 4.1 and 4.2. The algorithm was originally developed for standard hidden Markov fields referred to as HMF-IN (Section 2). The missing variables correspond to a Markov field on a discrete state space and the conditional independence assumption (2) is satisfied. We show below how algorithms developped for this HMF-IN case can be used to deal with more general models such as those in Section 3.

4.1 Learning step

We consider a supervised framework in which part of the information is available through learning data. It is assumed that for a number of individuals, we both observe x_i and its corresponding class y_i. Using the triplet model defined in Section 3, it remains that the z_i are missing. It follows that by considering variables $\mathbf{X}$ and $\mathbf{Z}|\mathbf{Y} = \mathbf{y}$, we can apply the algorithm described in Section 8.1 to the HMF-IN $(\mathbf{X}, \mathbf{Z}|\mathbf{Y} = \mathbf{y})$ (see equation (10)) to provide estimates of the model parameters which are the $\{\mathbb{B}_{ll'}, l, l' \in \{1, \ldots, L\}\}$ and the $\{\theta_{lk}, l = 1, \ldots L, k = 1, \ldots, K\}$. As mentioned in Section 3, estimating the later parameters is especially important to solve identifiability issues when dealing with our triplets Markov fields in the classification step. To estimate the θ_{lk}'s it is necessary that all L classes are sufficiently represented in the learning data. In practice the learning data are often divided in a number of separate data sets (*eg.* Section 7) so that the learning procedure actually consists of a number of separate runs of the estimation algorithm. As regards the $\mathbb{B}_{ll'}$'s estimated in the learning stage, we do not necessarily need to keep them for the classification step. However, for complex data, it may be that learning also the $\mathbb{B}_{ll'}$'s or at least part of them is a better choice in terms of modeling capabilities. We illustrate and explain such cases in more details in Section 7. This Section also presents a case where among the $\mathbb{B}_{ll'}$'s, only the $\mathbb{B}_{ll}$ can be learned due to the specificity of the learning data. Typically, if the underlying neighborhood structure is such that there exists no neighbors in classes l and l', then $\mathbb{B}_{ll'}$ cannot be estimated since terms involving $\mathbb{B}_{ll'}$ will not appear in the model formulas. More generally if the number of pairs in states l and l' is too small, the estimation of $\mathbb{B}_{ll'}$ is likely not to be good and in this case we should consider ignoring it.

When choosing to use in the subsequent classification step, all or part of the $\mathbb{B}_{ll'}$'s learned, considering separate runs for the estimation of the $\mathbb{B}_{ll'}$'s may raise identifiability issues. The model (9) used in each run is identifiable only up to a constant which may then vary from one run to another. The issue appears when grouping all estimations in a single model for the classification stage since various equivalent inferences under model (9) could lead to non equivalent inference under model (8). However, the explicit introduction of matrix $\mathbb{C}$ in (8) and the fact that its estimation is postponed to the classification step prevents this issue. Also, constraints on the form of $\mathbb{C}$ can be easily imposed (*eg.* Potts like constraint) to make the estimated parameters unique in the classification stage.

4.2 Classification step

At this stage, $\mathbf{Y}$ and $\mathbf{Z}$ are missing and only the observations $\mathbf{X}$ are available. Considering $\mathbf{X}$ and $\mathbf{U} = (\mathbf{Y}, \mathbf{Z})$, $(\mathbf{X}, \mathbf{U})$ is an HMF-IN (equations (7) and (6)) and we can apply again the algorithm of Section 8.1. The parameters are the $K \times K$ dimensional matrices $\{\mathbb{B}_{ll'},\ l, l' \in \{1, \ldots, L\}\}$ and the $\{\theta_{lk}, l = 1 \ldots L, k = 1 \ldots K\}$ as before with in addition the $L \times L$ dimensional matrix $\mathbb{C}$, *ie.* parameters $\{\mathbb{C}_{ll'},\ l, l' \in \{1, \ldots, L\}\}$.

The θ_{lk}'s are considered as fixed to the values computed in the learning stage. For the $\mathbb{B}_{ll'}$'s, different strategies arise depending on the available learning data and the goal in mind, in particular the type of interactions we want to account for. In practice, we propose to use specified or learned values for all $\mathbb{B}_{ll'}$'s. See Section 7 for an example. In most cases then, regarding parameters, the classification step consists of estimating $\mathbb{C}$. One possibility is to specify $\mathbb{C}$ to be of the Potts form *ie.* to consider diagonal $\mathbb{C}$, denoted by $\mathbb{C} = [\beta_l]$ when the diagonal terms are arbitrary, or $\mathbb{C} = [\beta]$ when they are all equal to some value β. More complex knowledge on the classes could be incorporated through other definitions of $\mathbb{C}$ but this simple case appears satisfying in a number of applications. Although $\mathbf{Y}$ is not Markovian, this acts as a regularizing term favoring homogeneous regions of the same class. This is an important feature of our classification step.

5 Selecting Triplet Markov Models

Choosing the probabilistic model that best accounts for the observed data is an important first step for the quality of the subsequent estimation and classification stages. In statistical problems, a commonly used selection criterion is the Bayesian Information Criterion (BIC) of [30]. The BIC is computed given the data $\mathbf{x}$ and a model $\mathcal{M}$ with parameters $\mathbf{\Psi}$. It is defined by:

$$BIC(\mathcal{M}) = 2 \log P(\mathbf{x} \mid \mathbf{\Psi}^{ml}) - \delta \log n \ ,$$

where $\mathbf{\Psi}^{ml}$ is the maximum likelihood estimate of $\mathbf{\Psi}$, $\mathbf{\Psi}^{ml} = \arg\max_{\mathbf{\Psi}} P(\mathbf{x} \mid \mathbf{\Psi}, \mathcal{M})$, δ is the number of free parameters in model $\mathcal{M}$ and n is the number of observations. BIC can be seen has an approximation of the probability of the model given the observations. The selected model is the one with the maximum BIC. BIC allows comparison of models with differing parameterizations. Many other approaches can be found in the literature on model selection (see for instance the list of references in [20]) but BIC has become quite popular due to its simplicity and its good results.

In this study, we consider the number of sub-classes (cardinality of the Z_i's state space, possibly varying as indicated in Section 3.2) as fixed to focus more specifically on the Markov model and on the Gaussian models. For the Markov model as defined in equations (7) and (8), model selection is done in two steps. We first select the best models for matrices $\mathbb{B}_{ll'}$'s . This can be done in the learning stage while finding the best model for matrix $\mathbb{C}$ can only be done in the test stage. We will in general only consider specific forms for $\mathbb{C}$ (Potts like). As regards matrices $\mathbb{B}_{ll'}$'s, omitting the subscripts, we will consider the decomposition of each of the $K \times K$ matrix into

$$\mathbb{B} = \Delta + \Gamma,$$

where similarly to Section 4, Δ is a diagonal matrix denoted by $\Delta = [\beta]$ if all diagonal terms are equal to a single value β and $\Delta = [\beta_k]$ if the diagonal terms are arbitrary. Conversely, for the second matrix Γ, all diagonal terms are 0. We then compare 4 possible models, namely, $\mathbb{B} = [\beta]$ (standard Potts model), $\mathbb{B} = [\beta_k]$ (generalized Potts model with class-dependent interaction parameters), $\mathbb{B} = [\beta] + \Gamma$ and $\mathbb{B} = [\beta_k] + \Gamma$ (unconstrained or full model).

For multivariate Gaussian sub-class specific distributions, there exists a number of different choices for the Σ_{lk}'s. See [1] and [10] for a description of examples of such forms and their mean-

ing. The simplest models are those for which the Σ_{lk}'s are diagonal matrices. We then compare this choice to the parameterizations described in Section 3.3 for high dimensional data.

However, for HMF's and for TMF's as well, the exact computation of BIC is not tractable due to the dependence structure induced by the Markov modeling. When focusing on the Gaussian parameters, a possibility is to compute BIC for independent mixture models, forgetting any spatial information but this would not make sense when choosing among various $\mathbb{B}$ models. We then propose to use a mean field like approximations of BIC proposed by [14] which is based on principles similar to that presented in Section 8.1. In what follows this approximated BIC will be denoted by BIC_{MF}. Examples of model selection results are shown in Section 7. Before that, as part of our experiments on simulated data, we mention and illustrate in the next section, a problem of phase transition, that can occur for the underlying Markov field (8), for its possible consequences on parameter estimation.

6 Simulated Triplet Markov fields and Phase transition phenomena

An important property associated with a Markov random field model is its ability to experience *phase transition*. This is a common physical phenomenon that occurs when a small change in physical parameters produces an abrupt and large-scale change in system properties. When Markov models are employed as a priori models for regularizing an inverse problem, such as image segmentation for example, then the performance of an algorithm that attempts to recover lost information from given data may strongly depend on the particular value of the parameters. The main reason for this is that most inverse algorithm attempt to recover information in a local and iterative fashion. This means that, at each iteration, the algorithm is only able to recover small-scale information. Large-scale information is recovered only after many iterations and as a consequence of the collective contribution of small-scale recoveries. It is therefore likely that depending on the location of the parameter in the parameter space, the inverse algorithm may suffer from critical slowing down. Determining the regions corresponding to phase transition and knowing the location of the parameters may then be important issues. In this section, we only illustrate such a phase transition using simulated data in a simple particular case of (8). A theoretical study on more general models is much more complex.

More specifically, phase transition corresponds to parameter values at which the partition function seizes of being analytic when the number of sites n grows to infinity [17].
Let us consider the Markov field $(\mathbf{Y}, \mathbf{Z})$ with pair potentials parameterized by $b, c \in \mathbb{R}$:

$$V_{ij}(z_i, y_i, z_j, y_j) = b z_i^t z_j \, y_i^t y_j + c y_i^t y_j \quad , \tag{13}$$

which is equation (8) with $\mathbb{B}_{ll'} = 0_L$ (the $L \times L$ zero matrix) if $l \neq l'$, $\mathbb{B}_{ll} = bI_K$ (where I_K denotes the $K \times K$ identity matrix) and $\mathbb{C} = cI_L$.

Figure 1 shows realizations of $(\mathbf{Y}, \mathbf{Z})$ for varying values of the two parameters and Figure 2 the corresponding realizations of $\mathbf{Y}$. Note that each of the 4 possible values of (y_i, z_i) is associated to a grey-level. Figure 3 shows simulated data $\mathbf{X}$ using a Triplet Markov models when the Gaussian distributions in 6 are one dimensional with standard deviation equal to 0.3. For comparison, Figure

4 shows realizations of the Gaussian HMF-IN models obtained using the images in Figure 2 and adding some Gaussian noise with 0 mean and standard deviation equal to 0.3. As can be suspected from Figure 1, such a simple case can illustrate a non trivial phase transition phenomenon. Let $W(b,c)$ be the partition function of $P(\mathbf{y},\mathbf{z})$. Second derivatives of W are given by:

$$\begin{aligned}
\frac{\partial^2 \log W(b,c)}{\partial b^2} &= Var(N(\mathbf{Y},\mathbf{Z})) \\
\frac{\partial^2 \log W(b,c)}{\partial c^2} &= Var(N(\mathbf{Y})) \\
\frac{\partial^2 \log W(b,c)}{\partial b \partial c} &= Cov(N(\mathbf{Y},\mathbf{Z}), N(\mathbf{Y}))
\end{aligned}$$

where $N(\mathbf{y},\mathbf{z})$ denotes the number of neighbors i and j such that $y_i = y_j$ and $z_i = z_j$, $N(\mathbf{y})$ the number of neighbors i and j such that $y_i = y_j$, Var the variance and Cov the covariance operator. Figure 5 is a superposition of the values of these derivatives for b and c varying between -2 and 2. These values were obtained by simulating different realizations of the Markov field (13). Different values of $N(\mathbf{Y},\mathbf{Z})$ and $N(\mathbf{Y})$ were then computed and the empirical variances and covariance of these quantities provided approximations of the derivatives. In Figure 5, a main "Y" shape curve, as well as 2 secondary curves indicate discontinuities in the partition function typical of critical parameter values. As an additional illustration, Figure 1 gives some realizations of the Markov field $(\mathbf{Y},\mathbf{Z})$ defined by equation (13) for $L = 2$ and $K = 2$.

7 Application to texture recognition with local invariant regions and their geometric relationships

The issue under consideration is a supervised clustering issue involving complex data. The data set is made of 140 single texture images and 63 multiple texture images. Images have been gathered over a wide range of viewpoints and scale changes. The data set contains $L = 7$ different textures illustrated in Figure 6. For each of the 7 textures, we have 20 single texture images from which 10 are kept for the learning set. The data set is then divided into a learning set containing 70 single texture images and a test set containing 70 other single texture images and 63 multiple texture images.

As mentioned before, traditional Gaussian MRFs modelling one dimensional grey-level intensity images cannot easily handle such viewpoint and scale varying images. More and more high-level image analysis, such as feature-based object recognition or object tracking, go beyond the traditional regular grids of pixels and one-dimensional grey-level intensities. Our images are then rather described by local affine-invariant descriptors and their spatial relationships. A graph is associated to an image with the nodes representing feature vectors describing image regions and the edges joining spatially related regions. Local photometric descriptors computed for invariant interest regions have proved to be very successful in applications such as object recognition, texture classification and texture recognition (see [6] and the references therein for preliminary work on such data). For the feature extraction stage, we follow the texture representation method described in [23] for its advantages over methods proposed in the recent literature. It is based on an interest point detector (Laplace

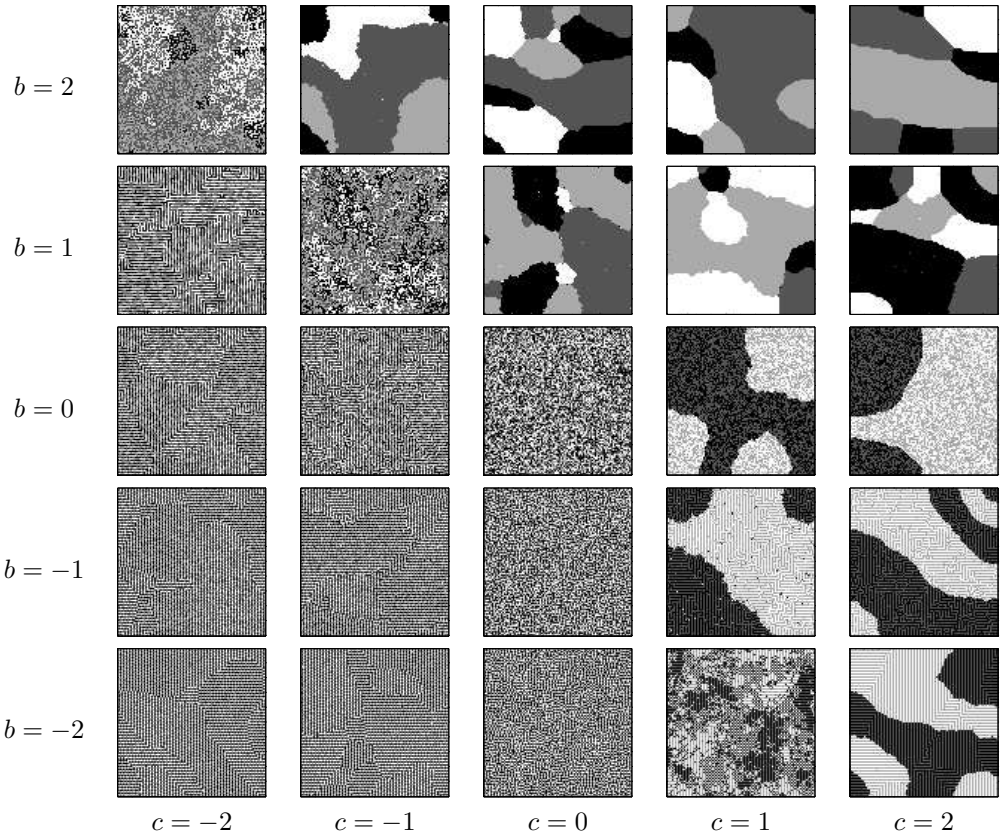

Figure 1: Realizations of $(\mathbf{Y}, \mathbf{Z})$ defined by (13) for various b and c, when $K = 2$ and $L = 2$.

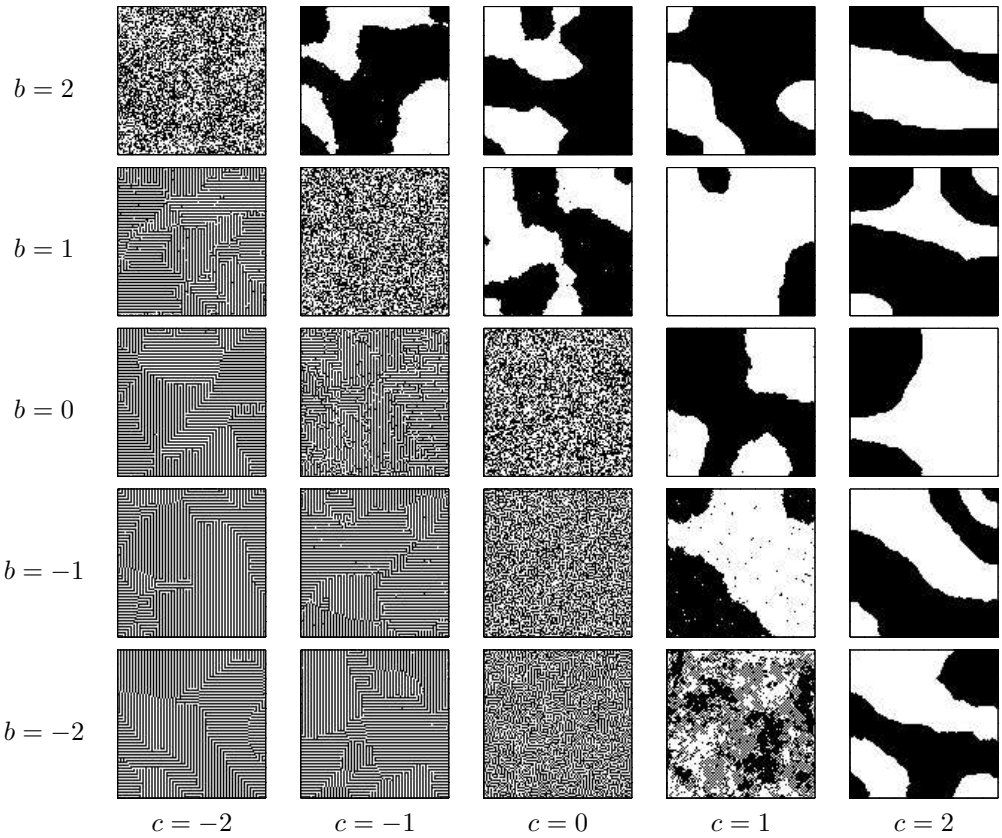

Figure 2: Realizations of $\mathbf{Y}$ for various b and c, when $K = 2$ and $L = 2$.

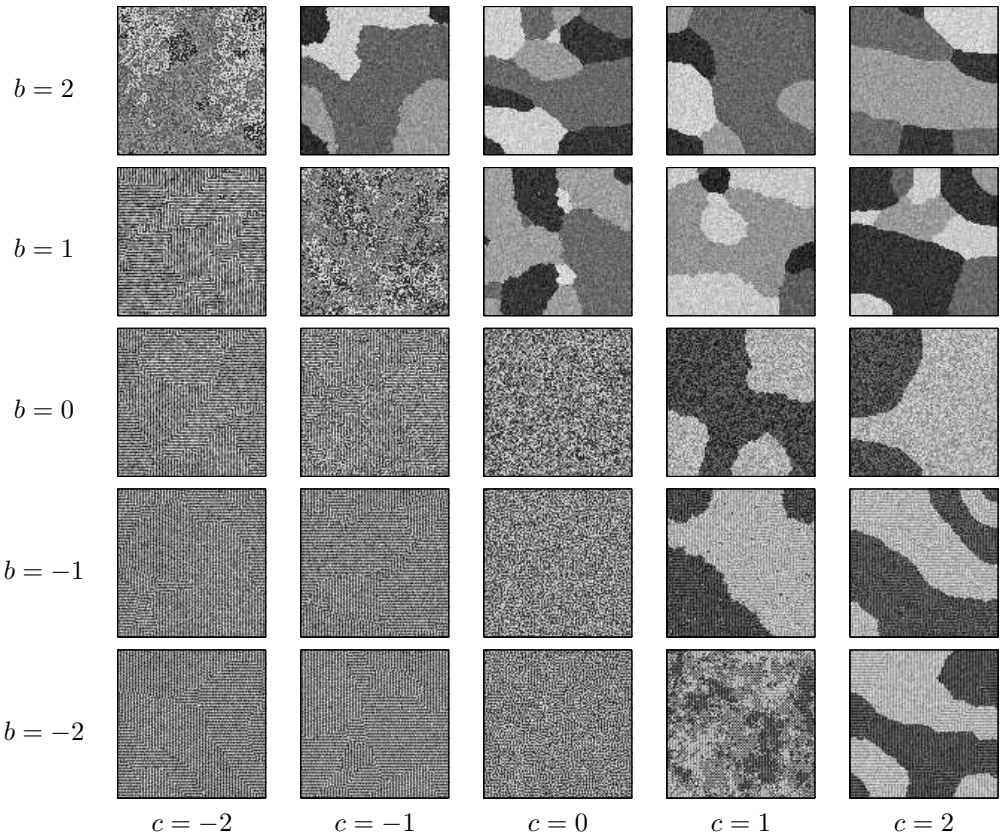

Figure 3: Realizations of $\mathbf{X}$ for various b and c, when $K = 2$ and $L = 2$.

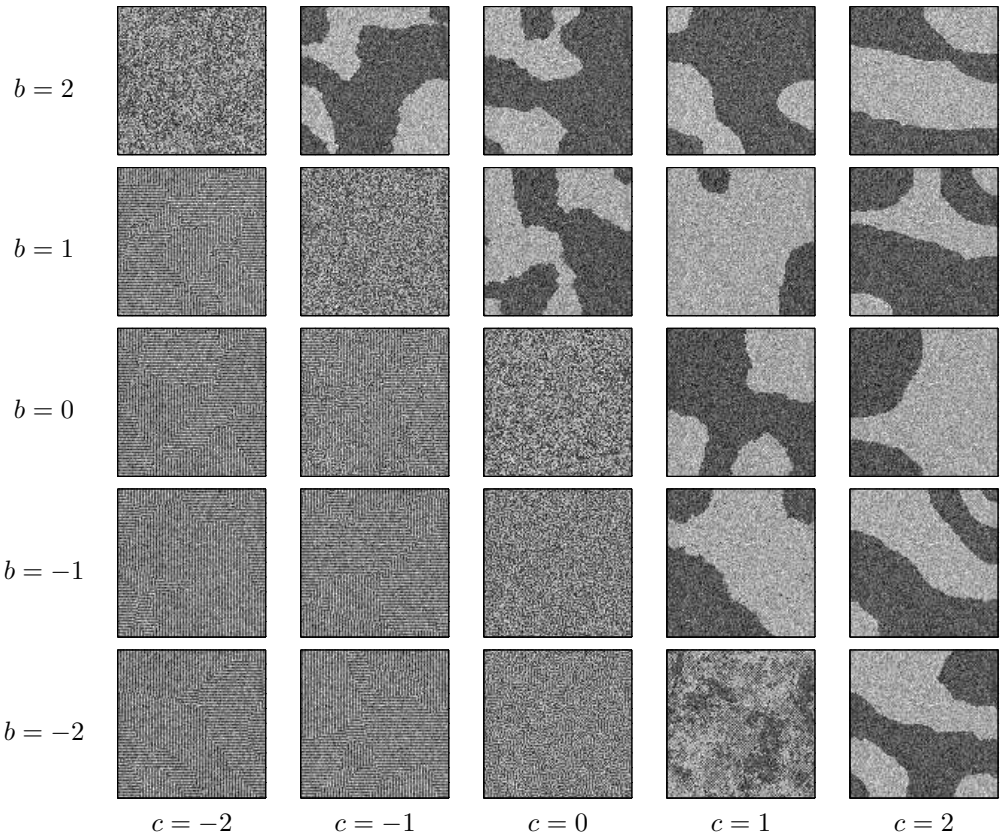

Figure 4: Realizations of a HMF-IN for various b and c, when $K = 2$ and $L = 2$.

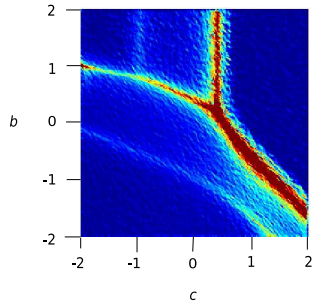

Figure 5: Phase transition for a simple 2 parameter model. A main "Y" shape curve, as well as 2 secondary curves, obtained by simulation, indicate where the discontinuities of the partition function are located and provide approximate values of the critical parameters.

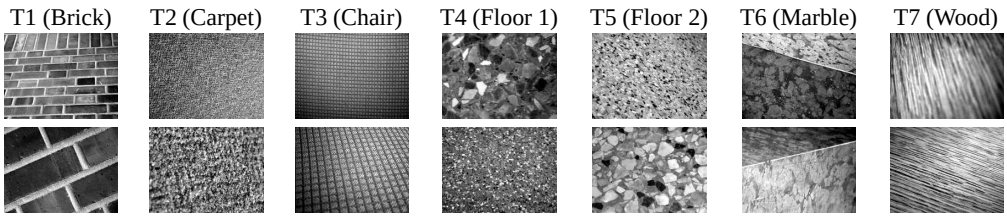

Figure 6: Samples of the texture classes used in the experiments.

detector) that leads to a sparse representation selecting the most perceptually salient regions in an image and on a *shape selection* process that provides affine invariance. Each detected region is then described by a feature vector (descriptor). The descriptors we use are 128-dimensional SIFT descriptors [27]. A graph is then built from the detected interest points by considering their Delaunay graph. Other choices, as regards graph construction and descriptors are possible [6]. In particular using graphs based on detected regions as in [23] is an issue when regions of very different sizes are detected. Delaunay graphs tend to provide more regular graphs where nodes all have a reasonable number of neighbors, with the possibility to put a threshold on too long edges. An illustration is given in Figure 7 that shows a multi texture image and the corresponding Delaunay graph. Beyond the actual choice of the interest point detector, feature vectors, etc. the specificities of this data set is the high dimensionality of the observed feature vectors, the irregularity of the sites at which they are observed and the non-unimodal nature of the class dependent distributions.

Our model assumes that descriptors are random variables with a specific probability distribution in each texture class. The number of sub-classes to described such distributions is set to $K = 10$ for each texture. Selecting K using BIC is also possible but we did not observe significantly better recognition results. For the 128-dimensional sub-class dependent Gaussian distributions, we consider two possibilities: diagonal covariance matrices or specific parameterization of the covariance

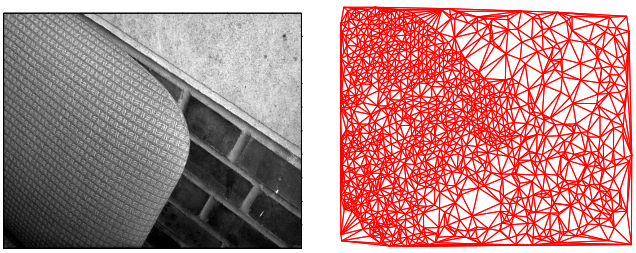

Figure 7: Multi texture image (left) and its associated Delaunay graph (right) built from the detected interest points.

matrices as described in Section 3.3. When dealing with high dimensional data, this reduces the number of parameters to be estimated significantly and tends to avoid numerical problems with singular matrices. As regards the Markov model, we consider that matrix $\mathbb{C}$ is fixed to a Potts form, *ie.* to a diagonal $[\beta]$ or $[\beta_l]$. Results are reported for the latter choice but the first one gives similar results. For matrices $\mathbb{B}_{ll'}$, the nature of the learning data set, including only single texture images, does not allow to estimate the $\mathbb{B}_{ll'}$'s for $l \neq l'$. We therefore set them to 0, which is consistent with the fact that we aim at recovering homogeneous regions of the same texture. An alternative is to postpone their estimation in the classification step but in practice test images do not generally include samples of all textures so that most of the $\mathbb{B}_{ll'}$ could not be estimated due to lack of relevant information. For the $\mathbb{B}_{ll}$'s we consider the possibilities described in Section 5 and use a mean field approximation of BIC to select the best model for each texture l with $l = 1, \ldots, L$. Again, the estimation of the $\mathbb{B}_{ll}$'s could be postponed to the classification step but this would mean estimating simultaneously on each test images, L matrices of size $K \times K$ (the bloc diagonal of a $KL \times KL$ dimensional matrix). Considering the number of detected points in each image (from few hundreds to few thousands), the estimation could be reasonably carried out only for very simple models such as diagonal models and would then greatly reduces the model flexibility. As an alternative, learning each texture separately involves less parameters and more data points allowing more complex models to be estimated accurately. Tables 1 and 2 report BIC_{MF} values for various models of $\mathbb{B}_{ll}$ in two cases corresponding to diagonal Σ_{lk}'s (Table 1) and the more general Σ_{lk}'s described in Section 3.3, referred to as *High Dim* Σ_{lk}'s. It appears that models with *High Dim* Σ_{lk}'s are always better, in terms of BIC_{MF}, whatever the $\mathbb{B}_{ll}$ model. For such Σ_{lk}'s (Table 2), the selected $\mathbb{B}_{ll}$ model depends on the texture class. It appears that for the wood texture, the simplest model is selected, while the more general model is selected only for the Floor 2 and Marble textures.

To illustrate and compare the various models performance, Table 3 shows recognition results for individual regions that is the fraction of all individual regions in the test images that were correctly classified. These results are obtained using only the single texture images. The "Mixture" columns refer to the method that assumes an independent Gaussian mixture for each image in the learning and classification steps. The two possible choices for the covariance matrices are considered. The EM algorithm is used for estimation and classification. The "TMF" columns refer to our method when

Diagonal Σ_{lk}'s	$\mathbb{B}_{ll}$ Model	Brick	Carpet	Chair	Floor 1	Floor 2	Marble	Wood
BIC_{MF}	$[\beta]$	1739730	2403890	3141300	2556280	3147610	2967630	2092090
	$[\beta_k]$	1739840	2403970	3141440	2556410	**3147900**	2967770	2092030
	$[\beta]$+Γ	1740010	2404450	3141930	2556630	3145570	**2968290**	1946380
	$[\beta_k]$+Γ	**1740080**	**2404600**	**3142170**	**2556730**	3145600	2968280	**2092760**

Table 1: $\mathbb{B}_{ll}$ model selection for each texture class when covariance matrices are assumed to be diagonal: the bold numbers indicate the model selected according to our BIC_{MF} criterion.

High Dim Σ_{lk}'s	$\mathbb{B}_{ll}$ Model	Brick	Carpet	Chair	Floor 1	Floor 2	Marble	Wood
BIC_{MF}	$[\beta]$	1902700	2515860	3516630	2696700	3290280	3172210	**2263160**
	$[\beta_k]$	1882040	**2524590**	**3529860**	**2697180**	3286890	3172450	2260870
	$[\beta]$+Γ	**1905800**	2518320	3521420	2692730	3292960	3177140	2260650
	$[\beta_k]$+Γ	1876510	2518430	3495890	2691310	**3293230**	**3178150**	2262300

Table 2: $\mathbb{B}_{ll}$ model selection for each texture class when covariance matrices are parameterized to account for high dimensional data: the bold numbers indicate the model selected according to our BIC_{MF} criterion.

Dependencie Model	Covariance Model	Brick	Carpet	Chair	Floor 1	Floor 2	Marble	Wood
Mixture	Diagonal Σ_{lk}	77.58	31.60	58.26	28.26	58.79	33.87	58.56
Mixture	High Dim Σ_{lk}	81.18	56.94	62.48	35.64	67.43	37.05	65.02
TMF	Diagonal Σ_{lk}	96.59	80.70	83.60	82.69	83.90	46.05	95.18
TMF	High Dim Σ_{lk}	99.33	98.61	99.28	97.36	99.57	56.24	99.28
TMF-BIC		**99.37**	**98.71**	**99.30**	**98.16**	**99.62**	**56.77**	**99.52**

Table 3: Percentage of individual regions correctly classified for the single texture images of the test set. Rows correspond to different models. The bold numbers indicate the higher percentages.

the more general model ([β_k]+Γ]) is used for all $\mathbb{B}_{ll}$'s with the two possible cases for the covariance matrices. The "TMF-BIC" column then refers to the case where the form of the covariance matrices and the $\mathbb{B}_{ll}$ models are selected according to BIC (Tables 1 and 2). The results in Table 3 show that the rates improve significantly on the independent Mixture rates (19% at the minimum) when our TMF model, with the *High Dim* parameterization of the Σ_{kl}'s, is used. For this latter case, the rates are all very good (98 % and above) except for the Marble texture. For this texture, the images available for learning are very heterogeneous in terms of lightning. On the same Marble image, some parts can be very badly lit and appear as very dark while others appear as very light. This prevent a good learning mainly due to the descriptor quality which cannot properly handle such variations.

For multiple texture images, significant improvement is also observed on all images. The rates increase about 53% in average between the *Mixture* and diagonal Σ_{lk}'s case and the *TMF-BIC* case. An illustration is given in Figure 8 with more details regarding the various possible models.

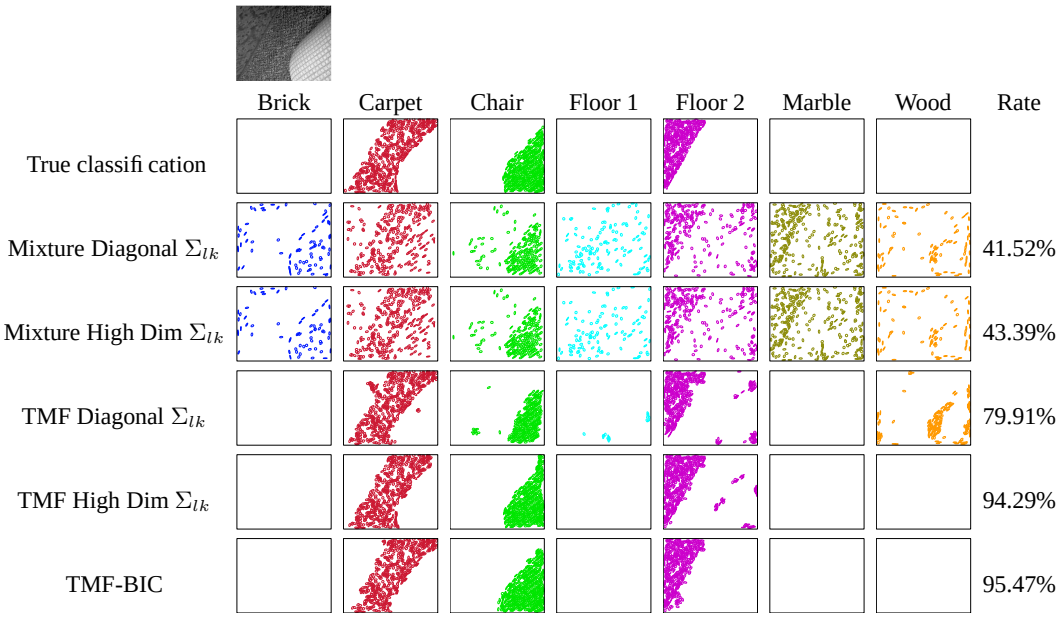

Figure 8: 3 texture (Carpet, Chair and Floor 2) image shown at the top-left corner: the first row shows the true classification while each following row corresponds to a different models. Columns show the interest points classified in each of the 7 texture classes. The last column reports the classification rate.

Rates for our TMF-BIC approach are all above 90 %. It happens in very few images that using TMF with the most complex $\mathbb{B}_{ll}$ models, instead of selecting them with BIC_{MF}, gives slightly better results (from 1 or 2%). Figure 10 illustrates such a case while Figure 9 illustrates the more general case where model selection leads to a significant gain.

As mentioned before (Table 3), the Marble texture suffers from lower recognition rates due to the nature of the learning data set. The high variability of the Marble images in this set makes learning a model for this texture very difficult. Figure 11 shows a typical example of a 3 texture image containing Marble to illustrate the behavior of our method in this case. Global recognition rates are still over 90 % but the errors mainly come from points in the Marble texture being misclassified. As regards the other methods, whose rates are not reported here, we observe similar results. Classification rates are greatly improved with our TMF-BIC method.

8 Discussion

We considered particular cases of Triplet Markov fields by designing them to include a learning stage and to adapt to general class dependent distributions or equivalently to general noise models.

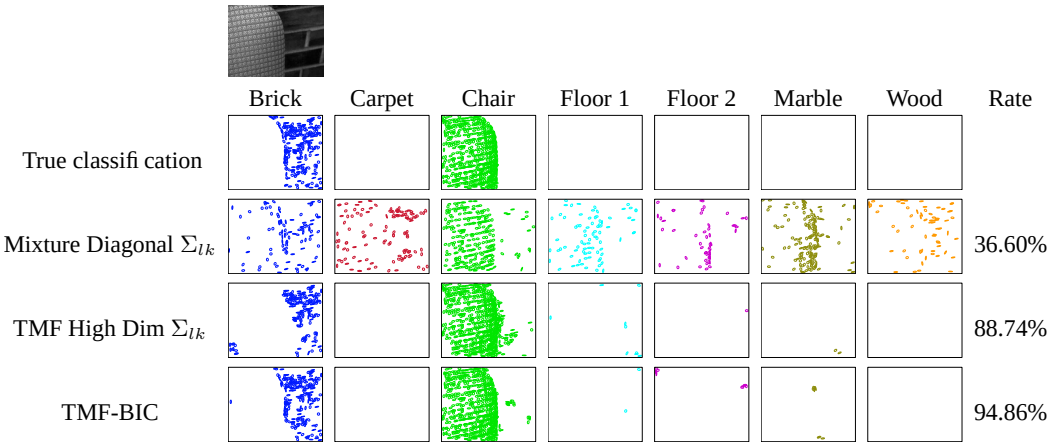

Figure 9: 2 texture (Chair and Brick) image shown at the top-left corner: the first row shows the true classification while the following rows correspond respectively to the independent mixture model with diagonal Σ_{lk}'s, the TMF model with *High Dim* Σ_{lk}'s and unconstrained $\mathbb{B}_{ll}$ models, the TMF model with selected $\mathbb{B}_{ll}$ models. Columns show the interest points classified in each of the 7 texture classes. The last column reports the classification rate.

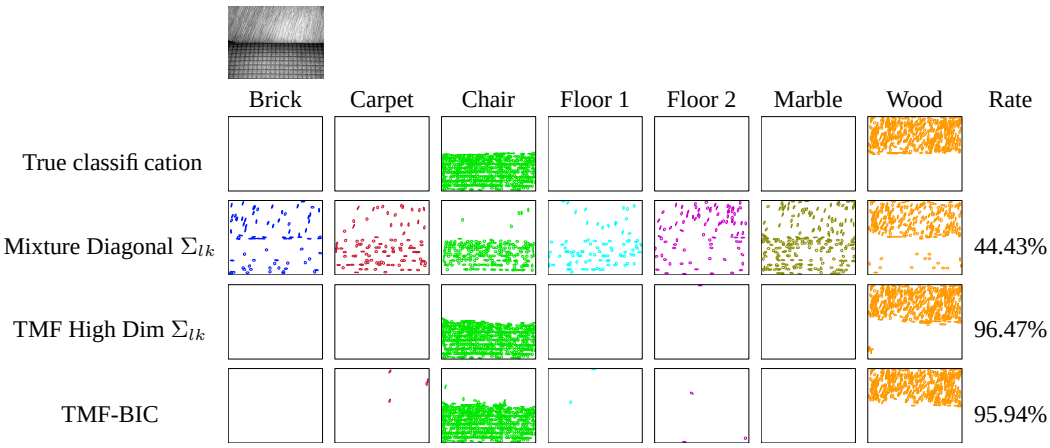

Figure 10: 2 texture (Chair and Wood) image shown at the top-left corner: the first row shows the true classification while the following rows correspond respectively to the independent mixture model with diagonal Σ_{lk}'s, the TMF model with *High Dim* Σ_{lk}'s and unconstrained $\mathbb{B}_{ll}$ models, the TMF model with selected $\mathbb{B}_{ll}$ models. Columns show the interest points classified in each of the 7 texture classes. The last column reports the classification rate.

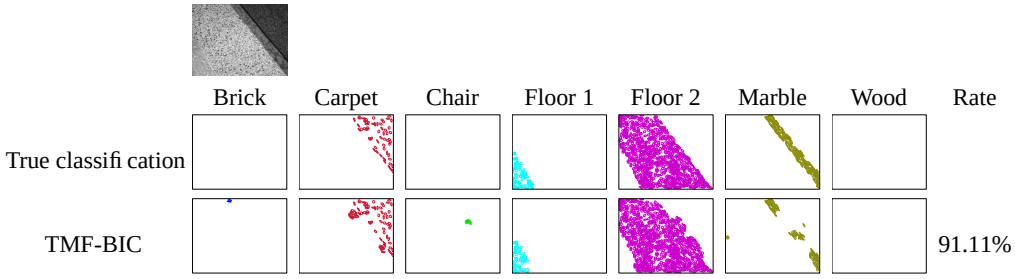

Figure 11: 3 texture image containing Marble shown at the top-left corner: the first row show the true classification and the second the classification obtained with our TMF-BIC method. Columns show the interest points classified in each of the 7 texture classes. The last column reports the classification rate.

Starting from a traditional hidden data model for which various estimation procedures exist, a *sub-class* variable $\mathbf{Z}$ is introduced in addition to the usual observed and missing variables $\mathbf{X}$ and $\mathbf{Y}$. The supervised problem is recasted as an unsupervised problem which allows traditional treatment. In particular our approach allows to model Markovian dependencies on the pixels and their effect on the noise parameter estimation. In a way similar to [2], introducing an extra $\mathbf{Z}$ allows to keep the same computational properties while increasing modelling capabilities.

The supervised framework was dictated by the type of applications in mind (it eg. texture recognition). The TMF model has shown its relevance in unsupervised frameworks too [2, 3] but our particular TMF's differ from the ones investigated in these papers in that some factorization properties do not hold. What limits our present study is the identifiability issue inherent to our model and the way we solve it by making full benefit of learning data. Alternative ways to deal with the identifiability issue in order to consider our TMF models in unsupervised cases would be interesting to investigate. This includes ideas related to the *relabelling algorithm* described in [19]. Our model is not limited to regular graphs. An interesting question that was not addressed in this paper involves the choice of the neighborhood structure. This choice may depend on the application. Indeed, for irregular lattices, the points relative displacements do not follow a predictable pattern and their linkage are not always obvious from their geometry so that a lot of possible spatial structures can be generated. As regards Markov models, the automatic neighborhood selection has not been really addressed in the literature except very recently by [24]. In our experiments, it appears that graphs with similar numbers of neighbors for each sites give more satisfying results. Directions of research for neighborhood selection can be found in [24].

Appendix

8.1 Parameter estimation procedure

In this section we describe the main features of the algorithm used for estimation in the two stages described in Sections 4.1 and 4.2. The algorithm was originally developed for standard hidden Markov fields referred to as HMF-IN (Section 2). To distinguish this particular case from the more general TMF cases considered above, we will denote by $\mathbf{O} = \{O_1, \ldots, O_n\}$ the observed variables and by $\mathbf{M} = \{M_1, \ldots, M_n\}$ the missing variables such that $(\mathbf{O}, \mathbf{M})$ is an HMF-IN, *ie.* $\mathbf{M}$ is Markovian on a discrete state space with G members $\{e_1, \ldots, e_G\}$ and the conditional independence assumption (2) is satisfied. As mentioned earlier, the learning stage (Section 4.1) is somewhat recasted as an unsupervised case so that the estimation procedures we consider are originally developed for unsupervised segmentation. We focus on soft clustering approaches and more specifically on EM based approaches. We consider recent procedures combining an EM approach with *mean field*-like approximations [9]. Such procedures were shown to be more efficient in many ways than standard Gibbs samplers or Markov Chain Monte Carlo (MCMC) techniques traditionally used in computer vision. The EM classification framework have many interesting features. As a probabilistic model, it leads to various possible statistical criteria to select automatically the number of clusters and it provides confidence measures such as posterior probabilities that an object (*eg.* a pixel) is assigned to a class. It is flexible in that various pairwise relationship information and features on individual data can be easily incorporated possibly with different weights. Its generalization to include missing data, that often occurs when dealing with real data, is straightforward and its extension to overlapping clustering methods, to deal with more realistic situations where objects can belong to many groups at the same time, can also be considered.

Briefly, these algorithms can be presented as follow (see [9]). They are based on the EM algorithm which is an iterative algorithm aiming at maximizing the log-likelihood (for the observed variables $\mathbf{o}$) of the model under consideration by maximizing at each iteration the expectation of the complete log-likelihood (for the observed and hidden variables $\mathbf{O}$ and $\mathbf{M}$) knowing the data and a current estimate of the model parameters. When the model is an Hidden Markov Model with parameters $\mathbf{\Psi}$, there are two difficulties in evaluating this expectation. Both the normalizing constant W in (1) and the conditional probabilities $P(m_i \mid \mathbf{o}, \mathbf{\Psi})$ and $P(m_i, m_j \mid \mathbf{o}, \mathbf{\Psi})$ for j in the neighborhood $N(i)$ of i, cannot be computed exactly. Informally, the mean field approach consists in approximating the intractable probabilities by neglecting fluctuations from the mean in the neighborhood of each site i. More generally, we talk about mean field-like approximations when the value for site i does not depend on the value for other sites which are all set to constants (not necessarily to the means) independently of the value for site i. These constant values denoted by $\tilde{m}_1, \ldots, \tilde{m}_n$ are not arbitrary but satisfy some appropriate consistency conditions (see [9]). Let $m_{N(i)}$ denote the set of variables $\{m_j, j \in N(i)\}$ associated to the set $N(i)$ of neighbors of i. It follows that $P(m_i \mid \mathbf{o}, \mathbf{\Psi})$ is approximated by

$$\begin{aligned} P(m_i \mid \mathbf{o}, \tilde{m}_{N(i)}, \mathbf{\Psi}) &\propto f(o_i|m_i^t\Theta)P(m_i|\tilde{m}_{N(i)}) \\ &\propto f(o_i|m_i^t\Theta)\exp(m_i^t(\mathbb{V}\sum_{j\in N(i)}\tilde{m}_j)) \end{aligned}$$

where Θ is considered as a vector of parameters. The normalizing constant is not specified but its computation is not an issue. Then, for all $j \in N(i)$, $P(m_i, m_j \mid \mathbf{o}, \mathbf{\Psi})$ is approximated by $P(m_i \mid \mathbf{o}, \tilde{m}_{N(i)}, \mathbf{\Psi})\ P(m_j \mid \mathbf{o}, \tilde{m}_{N(j)}, \mathbf{\Psi})$. Both approximations are easy to compute. Using such approximations leads to algorithms which in their general form consist in repeating two steps. At iteration q,

(1) Create from the data $\mathbf{o}$ and some current parameter estimates $\mathbf{\Psi}^{(q-1)}$, a configuration $\tilde{m}_1^{(q)}, \ldots \tilde{m}_n^{(q)}$. Replace the Markov distribution $P(\mathbf{m})$ defined as in (1) by the factorized distribution $\prod_{i=1}^{n} P(m_i|\tilde{m}_{N(i)}^{(q)})$.

It follows that the joint distribution $P(\mathbf{o}, \mathbf{m}|\mathbf{\Psi})$ can also be approximated by a factorized distribution:

$$\prod_{i=1}^{n} f(o_i|m_i^t\Theta)P(m_i|\tilde{m}_{N(i)}^{(q)})$$

and the two problems encountered when considering the EM algorithm with the exact joint distribution disappear. The second step is therefore, **(2)** Apply the EM algorithm for this factorized model with starting values $\mathbf{\Psi}^{(q-1)}$, to get updated estimates $\mathbf{\Psi}^{(q)}$ of the parameters.

In particular the *mean field* and *simulated field* algorithms consist in two different ways of performing step **(1)**. The *mean field* algorithm consists in updating the $\tilde{m_i}^{(q)}$'s by setting, for all $i = 1, \ldots, n$, $\tilde{m_i}^{(q)}$ to the mean of distribution $P(m_i \mid \mathbf{o}, \tilde{m}_{N(i)}^{(q)}, \mathbf{\Psi}^{(q-1)})$. Note that as M_i is an indicator vector, the mean value $\tilde{m_i}^{(q)}$ is a vector made of the respective probabilities to be in each of the G classes. In the *simulated field* algorithm, $\tilde{m_i}^{(q)}$ is simulated from $P(m_i \mid \mathbf{o}, \tilde{m}_{N(i)}^{(q)}, \mathbf{\Psi}^{(q-1)})$. Note also that to save additional notation, the updating described above is synchronous while we actually implemented a sequential updating of the $\tilde{m_i}^{(q)}$'s: each node i is updated in turn using the new values of the other nodes as soon as they become available rather than waiting until all nodes have been updated. Then, in practice, at step **(2)**, performing one EM iteration is usually enough. The HMRF estimation provides us with estimations for the means and covariance matrices of the G Gaussian distributions, namely μ_g and Σ_g for $g = 1, \ldots G$, but also for the hidden field parameters, matrix $\mathbb{V}$. It follows easily approximations of the $P(M_i = e_g|\mathbf{o}, \mathbf{\Psi})$'s required to classify each sites using the MPM or MAP principles.

References

[1] J. D. Banfield and A. E. Raftery. Model-based Gaussian and non Gaussian clustering. *Biometrics*, 49:803–821, 1993.

[2] D. Benboudjema and W. Pieczynski. Unsupervised image segmentation using Triplet Markov fields. *Comput. Vision Image Underst.*, 99:476–498, 2005.

[3] D. Benboudjema and W. Pieczynski. Unsupervised statistical segmentation of non stationary images using triplet Markov Fields. *IEEE Trans. PAMI*, 29(8):367–1378, 2007.

[4] J. Besag. On the statistical analysis of dirty pictures. *J. Roy. Statist. Soc. Ser. B*, 48(3):259–302, 1986.

[5] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[6] J. Blanchet, F. Forbes, and C. Schmid. Markov random fields for textures recognition with local invariant regions and their geometric relationships. In *BMVC*, Oxford, UK, September 2005.

[7] C. Bouveyron, S. Girard, and C. Schmid. High dimensional data clustering. *Comput. Statist. Data Analysis*, 2007. To appear.

[8] G. Celeux. Bayesian inference for mixtures: the label switching problem. In Physica-Verlag, editor, *COMPSTAT 98*, pages 227–232, 1998.

[9] G. Celeux, F. Forbes, and N. Peyrard. EM procedures using mean field-like approximations for Markov model-based image segmentation. *Pat. Rec.*, 36(1):131–144, 2003.

[10] G. Celeux and G. Govaert. Gaussian parsimonious clustering models. *J. of Pat. Rec. Soc.*, 28:781–793, 1995.

[11] B. Chalmond. An iterative Gibbsian technique for reconstruction of m-ary images. *Pat. Rec.*, 22(6):747–761, 1989.

[12] G. R. Cross and A. K. Jain. Markov Random Fields texture models. *IEEE trans. PAMI*, 5(1):2539, 1983.

[13] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B*, 39:1–38, 1977.

[14] F. Forbes and N. Peyrard. Hidden Markov random field model selection criteria based on mean field-like approximations. *IEEE trans. PAMI*, 25(8), 2003.

[15] O. Francois, S. Ancelet, and G. Guillot. Bayesian clustering using hidden Markov random fields in spatial genetics. *Genetics*, 174:805–816, 2006.

[16] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE trans. PAMI*, 6:721–741, 1984.

[17] J. Goutsias. Markov Random Fields: Interacting particle systems for image modelling and analysis. Technical report, Dept Electrical and Computer Engineering, Image Analysis and Communications Laboratory, The Johns Hopkins University, Baltimore, MD, USA, 1996.

[18] T. Hastie and R. Tibshirani. Dicriminant analysis by Gaussian mixtures. *J. Roy. Statist. Soc. Ser. B*, 58:158–176, 1996.

[19] A. Jasra, C. C. Holmes, and D. A. Stephens. Markov Chain Monte Carlo methods and the label switching problem in Bayesian Mixture modeling. *Stat. Sci.*, 20(1), 2005.

[20] R. Kass and A. Raftery. Bayes factor. *JASA*, 90:733–795, 1995.

[21] S. Kumar and M. Hebert. Discriminative random fields. *Int. J. Comput. Vision*, 68(2):179–201, 2006.

[22] S. Lauritzen, A. Dawid, B. Larsen, and H.-G. Leimer. Independence properties of directed Markov fields. *Networks*, 20:491–505, 1990.

[23] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. In *Proc. ICCV*, 2003.

[24] S. Le Hegarat-Mascle, A. Kallel, and X. Descombes. Ant colony optimization for image regularization based on a nonstationary Markov modeling. *IEEE Trans. IP*, 16(3):865–879, 2007.

[25] J. Li. Clustering based on a multi-layer mixture model. *J. Comput. Graph. Statist.*, 14(3), 2005.

[26] S. Z. Li. *Markov random field modeling in computer vision*. Springer-Verlag, London, UK, 1995.

[27] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[28] W. Pieczynski and A. Tebbache. Pairwise Markov Random Fields and segmentation of textured images. *Machine Graph. Vision*, 9:705–718, 2000.

[29] W. Qian and M. Titterington. Estimation of parameters in hidden Markov models. *Phil. Trans. R. Soc. Lond. A*, 337:407–428, 1991.

[30] G. Schwarz. Estimating the dimension of a model. *Ann. Stat.*, 6:461–464, 1978.

[31] M. Stephens. Dealing with label switching in mixture models. *J. Roy. Statist. Soc. Ser. B*, 62:795–809, 2000.

[32] V. N. Vapnik. *Statistical Learning Theory*. 1st edition, Wiley, 1998.

[33] M. Vignes and F. Forbes. Gene clustering via integrated Markov models combining individual and pairwise features. *IEEE trans. Comput. Biol. Bioinform.*, 2007. To appear.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifi que
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399