A tree-structured Markov random field model for Bayesian image segmentation.

Article *in* IEEE Transactions on Image Processing · January 2003

Source: DBLP

CITATIONS
116

READS
265

3 authors:

Ciro D'Elia
Università degli studi di Cassino e del Lazio Meridionale
34 PUBLICATIONS **395** CITATIONS
SEE PROFILE

Giovanni Poggi
University of Naples Federico II
111 PUBLICATIONS **1,725** CITATIONS
SEE PROFILE

Giuseppe Scarpa
University of Naples Federico II
72 PUBLICATIONS **948** CITATIONS
SEE PROFILE

A Tree-Structured Markov Random Field Model for Bayesian Image Segmentation

Ciro D'Elia, Giovanni Poggi, and Giuseppe Scarpa

***Abstract*—We present a new image segmentation algorithm based on a tree-structured binary MRF model. The image is recursively segmented in smaller and smaller regions until a stopping condition, local to each region, is met. Each elementary binary segmentation is obtained as the solution of a MAP estimation problem, with the region prior modeled as an MRF. Since only binary fields are used, and thanks to the tree structure, the algorithm is quite fast, and allows one to address the cluster validation problem in a seamless way. In addition, all field parameters are estimated locally, allowing for some spatial adaptivity. To improve segmentation accuracy, a split-and-merge procedure is also developed and a spatially adaptive MRF model is used.**

Numerical experiments on multispectral images show that the proposed algorithm is much faster than a similar reference algorithm based on "flat" MRF models, and its performance, in terms of segmentation accuracy and map smoothness, is comparable or even superior.

***Index Terms*—Markov random field (MRF), multispectral, remote sensing, segmentation, tree structure.**

I. Introduction

SEGMENTATION is the operation that allows one to extract the elementary homogeneous regions of an image. This low-level processing is of paramount importance for many subsequent applications. Remote sensing image classification [1], [2] is a good example: once the elementary regions are extracted, they can be classified, based on some relevant features, such as brightness, texture, shape, etc., as belonging to one of several known classes. Image and video compression [3] are other notable examples, where segmentation allows for the efficient representation of the objects that are present in a scene, their tracking and even their editing (as in the MPEG 4 standard). High-level processing tasks, related to image understanding and interpretation, always see at their core a segmentation process that raises the level of representation from pixel to region, in view of further levels of abstraction.

Barring trivial situations, segmentation is not an easy task. The very same concept of "homogeneous" region is not obviously defined, and a single textured region, readily identified by a human observer, could easily trick an automatic procedure and be erroneously partitioned in a large number of smaller regions. Even when there is no texture, segmentation errors can occur simply because of additive noise. To obtain acceptable results, one cannot rely solely on the observed data, but must take advantage of all available prior information about the image or class of images under analysis.

The Markov random field (MRF) model [4]–[8] is a relatively simple, yet effective, tool to encompass prior knowledge in the segmentation process, and in fact the interest on MRFs has been steadily growing in recent years. When image segmentation is formulated as a Bayesian, and more specifically a maximum *a posteriori* probability (MAP) estimation problem, all prior information available on the image X to be segmented, must be contained in its probability distribution $p(x)$. By modeling the image as an MRF, *i.e.*, assuming that each given pixel X_s depends statistically on the rest of the image only through a selected group of neighbors $X_{\eta(s)}$, one avoids the difficult problem of assigning a meaningful prior, and must only specify the local characteristics of the image $p(x_s|x_{\eta(s)})$. What is more important, local dependencies can be conveniently expressed through the definition of suitable potential functions in a Gibbs distribution. Of course, a number of problems remain open, the more important being

1) how to define an MRF (through its potentials) that is able to keep into account prior information while remaining mathematically and numerically manageable;
2) how to set/estimate the numerical parameters of such an MRF;
3) how to solve the MAP estimation problem with reasonable computational complexity.

The first problem is certainly the most intriguing, as it amounts to defining an abstract structure of the image that fits well the observed data. One could be tempted to define sophisticated models, in order to capture the complex nature of image dependencies. However, model definition cannot overlook the estimation problems (2) and (3). In fact, by increasing the model complexity one ends up with a large number of parameters that cannot be reliably estimated; and even neglecting this problem, the subsequent optimization task could be so computationally demanding to forbid the use of reliable procedures, leading to disappointing results. Indeed, computational complexity remains a major weakness of the MAP/MRF approach, and in developing a real-world MRF-based segmentation algorithm all efforts should be made to keep it under control, without sacrificing the description fidelity.

In recent years, a number of techniques have been proposed in order to obtain reliable MRF-based segmentation and classification with limited computational cost. In multi-resolution Bayesian segmentation, first proposed in [9], the basic structure

Manuscript received May 14, 2002; revised April 1, 2003. This work was supported in part by the Italian MIUR under Grant PRIN-2001. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mario A. T. Figueiredo.

The authors are with the Dipartimento di Ingegneria Elettronica e delle Telecomunicazioni, Università Federico II di Napoli, 21-80125 Napoli, Italy (e-mail: poggi@unina.it).

Digital Object Identifier 10.1109/TIP.2003.817257

is a tree, whose leaves are associated with the image pixels and whose intermediate nodes are associated with lower-resolution image cells. Each node of the tree depends only on a few nodes of the upper layer, up to the single root, and such dependencies along causal Markov chains allow for a much faster search of the optimal estimates. Major advances with respect to this basic idea have concerned the modeling of same-scale in addition to cross-scale dependencies [10]–[12], so as to improve segmentation quality (e.g., reduce blockyness). Another interesting approach is based on the use of region adjacency graphs (RAG) [13]–[15]. The basic idea is that regions that are clearly homogeneous should be treated as a whole in all processing steps and component pixels should not be individually analyzed. The immediate consequence is that only a relatively small number of regions must be dealt with, orders of magnitude less than the individual pixels, with a dramatic reduction in complexity. In more detail, after an initial over-segmentation of the image, carried out according to some suitable measure of homogeneity, the regions are considered as the nodes of a RAG on which a suitable MRF is defined and used for subsequent processing, such as merging, swaps, etc.

The very same goal of reducing complexity has led us to propose in [16] a constrained Markov random field model, based on a binary tree structure, where a K-class segmentation problem reduces to a sequence of $K-1$ much simpler binary segmentations. In such a model, named tree-structured Markov random field (TS-MRF), the whole image is associated with a tree of regions/segments, and each elementary region with a leaf in such a tree, which is progressively singled out by means of a sequence of binary decisions. As an example, to identify the "woods" region in the simple remote-sensing image shown in Fig. 1(a), one can first separate the "fields" region from the "river+woods" region, as shown in Fig. 1(b), and then, further separate "woods" from "river", Fig. 1(c). If the "fields" region is also segmented, the four-class image of Fig. 1(d) is finally obtained.

Tree-structured classification exhibits a number of additional advantages with respect to one-shot or "flat" classification [17]; let us briefly summarize the most important.

Speed. A binary segmentation is much simpler than a K-ary segmentation ($K > 2$). Moreover, in a tree-structured segmentation process, the image size keeps decreasing while descending the classification tree.

Adaptivity. Flat MRFs are usually homogeneous, meaning that their parameters do not adapt to local statistics of the image. Adaptive models can be defined [18], [19], but they tend to be unwieldy and very complex. In tree-structured segmentation, parameters adapt naturally to the local statistics of the image; for example, the "fields" region considered in Fig. 1 will have locally estimated parameters, that do not depend on the entire image.

Interpretability. As stated above, segmentation is often only the first step of more complex processes, such as classification, compression, image understanding. A flat segmentation provides little or no insight about the algorithm functioning, and even less space for interacting with it; on the contrary, tree-structured segmentation amounts to a sequence of binary decisions that are easily tracked, interpreted, and even guided by the end user if needed. A tree-structured segmentation map also allows for various types of image processing tasks [20], [21]. In addition, in the K-class segmentation map, a sequence of meaningful segmentation maps with $2, 3, \ldots, K-1$ classes is embedded, and the user can select the desired detail of description.

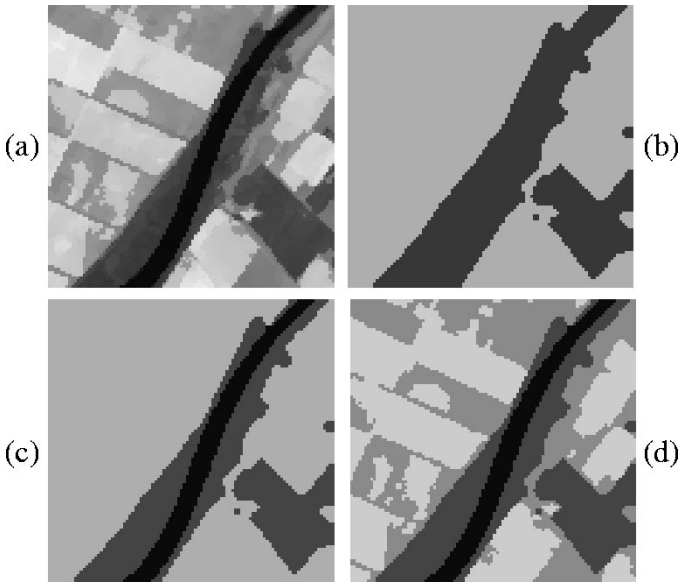

Fig. 1. Recursive binary segmentation. To extract the woods region, the original image (a) is segmented into two regions (b); then the darker region is further segmented (c) to separate "woods" from "river". If the lighter region in (c) is also segmented, map (d) is obtained.

It is worth pointing out that this tree-structured MRF is completely different from the multi-resolution tree-based models discussed before [9]–[12], where each node bears a label from the common state space $\{1, \ldots, K\}$. In our model, there is a progressive refinement in the class space for any single pixel, so that a pixel classified as "vegetation" at level one of the tree (to keep using a remote sensing example), could be more accurately classified as "crop" at level two, and as "soybean" at level three. In the multi-resolution models, instead, refinement occurs in resolution, so that a 16-pixel block, for example, classified as soybean at level one (a global "vegetation" class would not exist at all) is split in four 4-pixel blocks at level two, some classified as soybean and some, maybe, as corn, and each block is finally split in four isolated pixels to be classified again, as soybean or something else. Note also that the class-oriented and resolution-oriented hierarchies are not mutually exclusive, and could be used together when suitable.

In any case, a class-based hierarchy has a strong physical meaning because images are intrinsically tree-structured with component objects that are usually composed in turn of smaller units and so on in a possibly long chain [21]. This is not to say that a *binary* tree-structure is always adequate,[1] it is only a reasonable compromise between simplicity and accuracy. As a matter of fact, describing an image in terms of a binary-tree has its drawbacks, which obviously arise from the superimposition of a relatively rigid structure. The point is that, although any

[1]A tree with a variable number of children for each node would be fully general, thus including also the "flat" case.

number of component segments can be obtained through successive splits, each single split is constrained to be binary and therefore regions with different statistics (e.g, corn fields and woods) are temporarily grouped together. In the course of this process errors can occur, like splitting a single region between two nodes, and special attention must be devoted to make up for them. Also, if the segmentation tree is relatively deep, terminal nodes might be associated with regions composed of such a small number of pixels, that the correct estimation of relevant parameters becomes difficult.

In this work, we propose an image segmentation procedure based on the TS-MRF model that refines and improves upon the algorithm described in [16]. The growth of the segmentation tree is guided by a split gain that indicates whether or not a given node should be split, and measures the gain in description accuracy due to such a split. To make up for possible segmentation errors, originated by improper clustering and/or fragmentation of regions, we introduce a split-and-merge procedure, which allows us to modify dynamically the segmentation while building the tree. To this end a merge gain, specular to the split gain, is also defined and attached to couples of terminal nodes to guide the redefinition process. As said before, the use of tree-structured segmentation allows for a huge reduction of computational complexity with respect to methods based on flat MRF models. With the same goal in mind, the binary MRFs associated with each region are kept very simple, but we make them adaptive to counter the boundary fragmentation phenomenon, which occurs both with tree-structured and flat MRF models. As a result, simple and homogeneous segmentation maps are obtained.

Although the proposed segmentation algorithm is quite general, and should be applicable to any class of images, we focus attention here on remote-sensing multispectral images. To assess performance, we carry out a number of experiments with synthetic and real-world images, in order to test behavior both in a controlled environment, and with actual remote-sensing sources. Performance is measured in terms of segmentation fidelity (with objective measurements, when the ground truth is available) and computation time. In several situations, the proposed technique appears preferable to a comparable reference technique based on a flat MRF model not only for its reduced complexity, but also for its superior accuracy.

In the next section, we recall the basic definitions and concepts of MAP segmentation, MRF models and Gibbs distribution. Then, in Section III, the tree-structured MRF model is defined and the basic segmentation algorithm described. Section IV presents some experimental results in the segmentation of remote sensing multispectral images, which help identifying the weak points of the algorithm. In Section V, an improved segmentation algorithm is proposed, where segment reshaping is allowed and boundary fragmentation is avoided. Section VI assesses the algorithm performance through numerical experiments on real-world and synthetic data. Finally, Section VII draws conclusions, describes open problems and outlines future research.

II. Image Segmentation Based on MRF Models

A digital image defined on a discrete set of sites $\mathcal{S}$ (not necessarily a rectangular lattice) can be conveniently modeled as a random field, namely, a collection of random variables, one for each site, $X = \{X_s, s \in \mathcal{S}\}$ with realization $x = \{x_s \in \Omega_s, s \in \mathcal{S}\}$, and a valid joint probability distribution. Whenever unambiguous, we will indicate the probability law associated with X simply as $p(x)$, to be meant as either a density or a distribution function depending on the case.

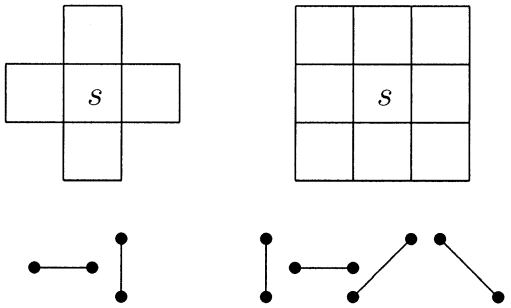

Fig. 2. Pixel s and its neighbors in the η^1 (left) and η^2 (right) system. In the η^2 system diagonal two-site cliques can be used.

Through the use of random fields, one can treat segmentation as a statistical estimation problem in the presence of noise. The observed image, say Y, is modeled as the sum[2] of an underlying "true" image and a noise process $Y_s = g(X_s) + N_s$. X_s can assume only a finite number of values, or labels $x_s = 1, 2, \ldots, K$, and $g(\cdot)$ performs a univocal mapping from the label space to the space of observed data, so that $g(x_s)$ is dimensionally homogeneous with y_s and n_s.

The goal of segmentation is to estimate the correct label for each s. A quite common approach is to minimize the probability of error given the data y, and hence resort to MAP (maximum *a posteriori* probability) estimation:

$$\widehat{x} = \arg\max_x p(x|y) = \arg\max_x p(y|x)p(x). \tag{1}$$

The likelihood term $p(y|x)$ depends only on the noise distribution, assumed to be perfectly known for the time being, and hence only two major problems remain: appropriately modeling the prior image distribution $p(x)$ with its parameters, and finding the maximum in (1).

As was said in the Introduction, the modeling problem can be much simplified by assuming that the label image X is fully characterized by local dependencies. In particular, if the Markovian property holds true for each site s, namely, X_s is conditionally independent of the rest of the image given the realization on a neighborhood $\eta(s)$

$$p(x_s|x_{\mathcal{S}-s}) = p(x_s|x_{\eta(s)}) \tag{2}$$

then X is said to be an MRF with respect to the neighborhood system $\eta = \{\eta(s), s \in \mathcal{S}\}$. Quite often, in order to limit complexity, only the 4 or 8 closest pixels (system η^1 and η^2, respectively) are included in a pixel's neighborhood (see Fig. 2). Even so, the MRF model proves quite powerful because most of the dependencies can be captured through local interactions.

[2]Nonlinear combinations and blurring are usually not considered in this context.

What makes MRFs especially attractive is the Hammersley-Clifford theorem [7] which proves that any positive MRF has a Gibbs probability law

$$p(x) = \frac{1}{Z}\exp[-U(x)] = \frac{1}{Z}\exp\left[-\sum_{c\in\mathcal{C}} V_c(x)\right] \quad (3)$$

where the $V_c(\cdot)$ functions are called potentials, $U(x)$ is named energy, and $Z = \sum_x \exp[-\sum_c V_c(x)]$ is just a normalizing constant.[3] In (3), c indicates a clique of the image, which is a set of sites that are all neighbors of one another (see again Fig. 2). Note that each potential V_c depends only on the values taken on the clique sites $\{x_s, s \in c\}$ and, therefore, accounts only for local interactions. As a consequence, local dependencies in X can be easily modeled by defining suitable potentials $V_c(\cdot)$, as shown for example in [7]. With the MRF-Gibbs equivalence, the model selection problem amounts to choosing a suitable neighborhood system and suitable potentials so as to express the desired spatial dependencies. A number of models have been proposed in the literature for various applications and we refer the reader to [8] for a comprehensive review.

Assuming that one has been able to select a satisfactory model for the prior distribution, the problem remains of maximizing $p(x)p(y|x)$ over x, which is when computational complexity becomes a dominant concern. It can be shown that $p(x)p(y|x)$ is itself a Gibbs distribution, and its maximization can be carried out via the simulated annealing (SA) technique [5], but only under conditions (a very slow cooling schedule) that make it unfeasible in practice. Using a faster schedule, on the other hand, compromises SA's optimality. Gradient-based algorithms tend to remain trapped in local optima as the objective function is in general nonconvex. This is the case of the iterated conditional modes (ICM) algorithm [7], structurally similar to the SA but much faster, which is why it is often the algorithm of choice. Even resorting to the ICM, and hence renouncing global optimality, computational complexity remains the major bottleneck of MRF-based segmentation, and the problem only worsens when there is no *a priori* knowledge on the image, namely in unsupervised segmentation.

Quite often, indeed, a number of important parameters such as the number K of labels/classes in the image, the parameters of the likelihood term $p(y|x)$, and the parameters of the Gibbs prior $p(x)$, are not known, and must be estimated from the data together with the segmentation $\widehat{x}$ itself. The single most critical parameter is by far the number of classes K, since it influences heavily all other aspects of segmentation.

The problem of determining the number of classes in a data set, or cluster validation problem, has received much attention in the literature [22], with mixed and inconclusive results. As a matter of fact, in a real-world image, the number of different segments that can be identified varies wildly according to the user's point of view. In a remote-sensing image, for example, a single segment labeled as "urban area" in one application, could be further partitioned in smaller segments in another application. In the absence of prior information on the application, both solutions are equally reasonable, and both should be preserved to let a human interpreter have a final say. Although some efficient strategies have been proposed to address the cluster validation problem, especially in the context of split-and-merge approaches, this is still one of the main reasons for the increase in complexity going from supervised to unsupervised segmentation.

Another reason is the need to estimate, together with the segmentation, the parameters of the involved distributions, collectively represented by a random vector Θ

$$(\widehat{x}, \widehat{\theta}) = \arg\max_{x,\theta} p(x, y|\theta). \quad (4)$$

Since exact joint optimization is computationally intractable, a two-step procedure is often used. First, the model parameters are estimated from the observed data, following for example an ML approach, then the MAP segmentation is carried out in a second step using the estimated parameter values. A number of techniques can be used to perform the ML parameter estimation, such as the EM algorithm [23] and its numerous variants, or the similar but more general ICE [24]. Except for some simple cases, however, these algorithms do not have an analytical closed form, and are quite computationally expensive. For this reason we will resort to the suboptimal, but much simpler, alternating marginal optimization ($\widehat{x}$ and $\widehat{\theta}$ are alternately optimized given each other) which can be viewed as an approximation of the two step EM-approach, and has been observed to provide comparable results in various practical situations [25].

III. Tree-Structured MRF's and Image Segmentation

Our interest for tree-structured segmentation originates with work in the image compression field [26]. Indeed, links between data compression and segmentation are very tight, and their mutual interactions have been only partially explored to date [27]. Vector quantization, for example, can be seen both as a data compression and a data segmentation algorithm, and tree-structured vector quantization (TSVQ) is a well-known and widely used low-complexity compression technique, that has close ties with classification and regression trees [17], [28]. In [26] we resorted to TSVQ as the basic tool for a segmentation-based image compression algorithm. The use of a preliminary segmentation allowed us to apply a region-adaptive Karhunen-Loeve transform, thus improving the energy compaction ability of the transform and, eventually, the overall compression performance. So, replacing TSVQ with segmentation based on a tree-structured MRF model was an obvious step to improve segmentation and compression performances at once by taking into account spatial dependencies at a limited cost.

The algorithm discussed here, and proposed in its basic form in [16], is based on the recursive binary segmentation of image regions. The whole image is modeled originally as a binary MRF and segmented accordingly. Then, all newly generated segments are modeled in their turn as binary MRFs and possibly segmented, and this process goes on recursively until some stopping condition is met. The choice of *binary* partitions reduces dramatically the computational complexity, however, it can give rise to segmentation inaccuracies, that must be addressed, as shown in Section V, to obtain fully satisfactory results.

[3] Often, a temperature parameter appears in the definition, $p(x) = Z^{-1}\exp[1/T\sum_c V_c(x)]$, but we include it in the potentials since it is not explicitly needed in the following.

A. The TS-MRF Model

Let us consider a binary tree T, identified by its nodes and their mutual relationships. Except for the root, each node t has one parent $u(t)$, and each internal node has two children $l(t)$ and $r(t)$, with $u[l(t)] = u[r(t)] = t$.[4] We also define $\widetilde{T} = \{t \in T : l(t) = r(t) = \emptyset\}$, the set of terminal nodes or leaves, and $\overline{T} = T - \widetilde{T}$, the set of internal nodes. With each node t in T we associate the following items:

- a set of sites $\mathcal{S}^t \subseteq \mathcal{S}$, corresponding to a segment of the image (in particular $\mathcal{S}^{\text{root}} = \mathcal{S}$);
- a binary MRF $X^t = \{X_s^t : s \in \mathcal{S}^t\}$, with neighborhood system η^t, and realization x^t where $x_s^t \in \{l(t), r(t)\}$;
- a set of parameters θ^t that specify the potentials $V_c^t(\cdot)$ of the Gibbs distribution $p(x^t) = p(x^t; \theta^t)$.

Now we impose the additional constraint that the set of sites associated with any given node is obtained from the binary segmentation of the parent set of sites. More formally, for each internal node of the tree $t \in \overline{T}$

$$\begin{cases} \mathcal{S}^{l(t)} &= \{s \in \mathcal{S}^t : x_s^t = l(t)\} \\ \mathcal{S}^{r(t)} &= \{s \in \mathcal{S}^t : x_s^t = r(t)\} \end{cases}. \tag{5}$$

Given this condition, we define the tree-structured MRF X^T associated with the tree T, as the set of all binary fields associated with the internal nodes of T

$$X^T = \bigcup_{t \in \overline{T}} X^t \tag{6}$$

with realization $x^T = \bigcup_{t \in \overline{T}} x^t$ and distribution law $p(x^T) = p(x^t, t \in \overline{T})$.

As an example, consider the tree T depicted in Fig. 3(a), with the interior nodes $\overline{T} = \{1, 2\}$ shown as empty circles, and the leaves $\widetilde{T} = \{3, 4, 5\}$ as filled circles. The TS-MRF associated with T is $X^T = \{X^1, X^2\}$, and its realization $x^T = \{x^1, x^2\}$ has probability distribution

$$p(x^T) = p(x^1, x^2) = p(x^1)p(x^2|x^1) \tag{7}$$

X^1 is a binary Markov random field defined on the set of sites $\mathcal{S}^1 = \mathcal{S}$, with neighborhood system η^1 (the system associated with node 1, not to be confused with the first-order system) and potentials $V_c^1(\cdot)$, defined on the cliques $c \in \mathcal{C}^1$, where $\mathcal{C}^1$ is the set of cliques of interest for X^1. Therefore

$$p(x^1) = \frac{1}{Z^1} \exp\left[- \sum_{c \in \mathcal{C}^1} V_c^1(x^1) \right]. \tag{8}$$

By definition $x_s^1 \in \{l(1), r(1)\} = \{2, 3\}$ for each s in $\mathcal{S}^1$, so the realization x^1 partitions the original set of sites in two subsets $\mathcal{S}^2$ and $\mathcal{S}^3$, according to (5). As an example, consider again the image of Fig. 1(a), associated with the rectangular set of sites $\mathcal{S}^1$. Its segmentation generates two subimages, shown in Fig. 4, on each of which a new MRF can be defined.[5] In particular, we

[4]Although not necessary, it is common to associate nodes with integers, so that $\text{root} = 1$, $l(t) = 2t$, $r(t) = 2t + 1$ and $u(t) = \lfloor t/2 \rfloor$. We will follow this convention whenever possible in order to simplify notation.

[5]Note that the new set of sites are not necessarily connected, as is clear by this example.

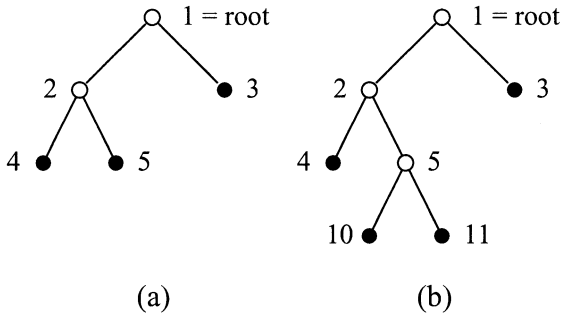

Fig. 3. (a) Simple binary tree and (b) the tree resulting from the split of node 5.

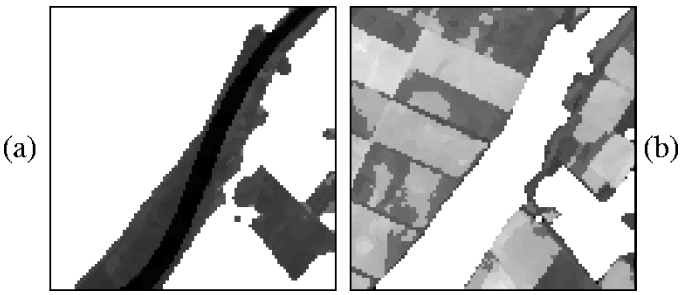

Fig. 4. Binary segmentation of the image in Fig. 1(a) generates two subimages, shown in parts (a) and (b), that can be further segmented.

define X^2 on $\mathcal{S}^2$ with its specific attributes η^2, $\mathcal{C}^2$, $V_c^2(\cdot)$. Its distribution, conditional on x^1, will therefore be

$$p(x^2|x^1) = \frac{1}{Z^2(x^1)} \exp\left[- \sum_{c \in \mathcal{C}^2(x^1)} V_c^2(x^2) \right] \tag{9}$$

where the set of valid cliques for X^2 is composed of all cliques of η^2 such that all their sites belong to $\mathcal{S}^2$. Therefore, the conditioning on the father MRF operates on the selection of the set of sites $p(x^t|x^{u(t)}) = p(x^t|\mathcal{S}^t)$ and, consequently, on the local partition function. Note that, just because of the dependence of Z^t on $x^{u(t)}$, the TS-MRF is *not* equivalent to a flat MRF defined on the neighborhood system $\eta^T = \bigcup_{t \in \overline{T}} \eta^t$, even though each component field X^t is Markovian on η^t.

TS-MRFs bear some resemblance with the hierarchical Markov random fields (H-MRF) proposed to model and analyze textured images [8]. However, the structure of H-MRF has just two levels, the first level segments the image in K blob-like regions according to large scale properties (like the average intensity), while the second level models each region's interior as a nonisotropic MRF in order to account for local patterns at a finer scale. Similarities are not striking: H-MRF is essentially a modeling tool and, more important, it does not reduce computational complexity as a K-ary segmentation is still required.

B. Unsupervised Image Segmentation Based on a TS-MRF Model

The proposed segmentation algorithm has a recursive nature, it starts with a single-node tree which grows leaf by leaf until a stopping condition is met. Therefore, we first describe the algorithm initialization, focusing on the root (node 1), and then the generic step with reference to a given tree.

At the beginning we consider the following two hypotheses:

$$\begin{cases} H_0: & T=\{1\}, & X^T=\emptyset \\ H_1: & T'=\{1,2,3\}, & X^{T'}=\widehat{x}^1 \end{cases}. \tag{10}$$

The first hypothesis corresponds to the case in which the whole image, associated with the root node ($\mathcal{S}^1=\mathcal{S}$, $y^1=y$), is represented as a single region. Therefore, the observed data are described by a single distribution $p(y^1)$, whose model is known but for some parameters ν^1 that must be estimated from the data themselves. Of course, in this case the TS-MRF is empty, and all sites have the same label attached. This is the only possible configuration and in this sense we define $p(x^{\{1\}})=1$, and also write the data distribution as $p(y^1|\nu^1)$ to make explicit that y^1 is described through the single set of parameters ν^1 attached with node 1.

The second hypothesis corresponds to the case in which the image is represented by two regions. To single out such regions, a binary MRF X^1 is defined on $\mathcal{S}^1$, with a given neighborhood system η^1, and with potentials $V_c^1(\cdot)$ that are completely specified except for some parameters θ^1. The MAP estimate of the MRF $\widehat{x}^1$, with probability $p(\widehat{x}^1)$, divides the image into two new regions, $\mathcal{S}^2=\{s\in\mathcal{S}^1:\widehat{x}_s^1=2\}$ and $\mathcal{S}^3=\{s\in\mathcal{S}^1:\widehat{x}_s^1=3\}$, with their associated data y^2 and y^3. We assume, as is usually done, that the data are conditionally independent given the labels, and hence their description factors out as $p(y^1|x^1)=p(y^2|\nu^2)p(y^3|\nu^3)$.

At this point, we compare the two statistical descriptions of the image, based on a single-class model (tree T) or a two-class model (tree T'), by checking the condition

$$G^1=\frac{p(\widehat{x}^{T'})p(y|\widehat{x}^{T'})}{p(\widehat{x}^T)p(y|\widehat{x}^T)}>1 \tag{11}$$

which, specialized for $T=\{1\}$, becomes

$$G^1=\frac{p(\widehat{x}^1)}{1}\times\frac{p(y^1|\widehat{x}^1)}{p(y^1|\nu^1)}>1. \tag{12}$$

If the test succeeds, namely the split gain G^1 is greater than 1, the two-region description better fits the data and the procedure goes on, otherwise it stops and the single-region description is accepted.

Let us now consider a generic tree T, that has been temporarily accepted as our structure, with associated TS-MRF X^T, and let τ be a leaf of T that we are testing for a possible split. The two hypotheses under test are then

$$\begin{cases} H_0: T, & X^T=\widehat{x}^T \\ H_1: T'=\mathrm{split}(T,\tau), & X^{T'}=(\widehat{x}^T,\widehat{x}^\tau) \end{cases} \tag{13}$$

where tree $T'=\mathrm{split}(T,\tau)$ is identical to T except for node τ which generates two new leaves, 2τ and $2\tau+1$, becoming itself an internal node [[see Fig. 3(b)]. To explicit test (11) for the general case, remember that $p(\widehat{x}^T)=p(\widehat{x}^1)\prod_{t\in\overline{T},t\neq 1}p(\widehat{x}^t|\widehat{x}^{u(t)})$, and that $p(\widehat{x}^t|\widehat{x}^{u(t)})=p(\widehat{x}^t|\mathcal{S}^t)$. Moreover $p(y|\widehat{x}^T)=\prod_{t\in\widetilde{T}}p(y^t|\nu^t)$. Therefore, we can write

$$\begin{aligned} p(\widehat{x}^T)&=\prod_{t\in\overline{T}}p(\widehat{x}^t|\mathcal{S}^t)\\ p(\widehat{x}^{T'})&=\prod_{t\in\overline{T}'}p(\widehat{x}^t|\mathcal{S}^t)=p(\widehat{x}^\tau|\mathcal{S}^\tau)\prod_{t\in\overline{T}}p(\widehat{x}^t|\mathcal{S}^t)\\ p(y|\widehat{x}^T)&=\prod_{t\in\widetilde{T}}p(y^t|\nu^t)=p(y^\tau|\nu^\tau)\prod_{t\in\widetilde{T},t\neq\tau}p(y^t|\nu^t)\\ p(y|\widehat{x}^{T'})&=\prod_{t\in\widetilde{T}'}p(y^t|\nu^t)=p(y^\tau|\widehat{x}^\tau)\prod_{t\in\widetilde{T},t\neq\tau}p(y^t|\nu^t) \end{aligned} \tag{14}$$

and the test becomes simply

$$G^\tau=\frac{p(\widehat{x}^\tau|\mathcal{S}^\tau)}{1}\times\frac{p(y^\tau|\widehat{x}^\tau)}{p(y^\tau|\nu^\tau)}>1. \tag{15}$$

It should be noted that the test depends exclusively on region $\mathcal{S}^\tau$. In fact, given $\widehat{x}^T$ the maximization process operates only on x^τ, and the MAP problem reduces to

$$\widehat{x}^\tau=\arg\max_{x^\tau}p(x^\tau)p(y^\tau|x^\tau) \tag{16}$$

completely local to node τ. If the test succeeds, the growth of the tree and of the associated segmentation continues in a similar way for each newly created leaf, as if each one were the root of a new tree. Therefore, the tree growing process is accurately described by a recursive procedure, which can go on in parallel for each node.

The ratio G^τ, named split gain, accounts for the gain in description efficiency arising from the split of leaf τ. This interpretation becomes more compelling if we take the logarithm of G^τ and regard it as the difference $\log G^\tau=I(T)-I(T')$ between the self-information associated with each of the competing TS-MRFs.[6] If the self-information is a good indicator of the description complexity, then a positive log split gain indicates that the new description of the observed data is "simpler" than the preceding one, and hence preferable (according to Occam's razor). In more detail, a split has always a cost, $p(\widehat{x}^t)<1$, due to the need of describing the segmentation $\widehat{x}^t$, but also a value, $p(y^t|\widehat{x}^t)/p(y^t|\nu^t)>1$, because the data are more accurately represented, in each new segment, by their local parameters. A positive $\log G^t$ indicates that the overall benefits outweigh the cost. It must be underlined, however, that the evaluation of the split gain involves an intractable partition function and that only an approximation of it, possibly inaccurate, will be available in any practical implementation.

Fig. 5 shows a high-level flow chart of the segmentation algorithm. To improve readability, the procedure is sequential rather than recursive, and only one leaf at a time is split, the one with the largest split gain (the experiments will follow this convention as well).

- In the initialization step, the tree is defined as consisting of the sole root ($T=\{1\}$); the whole image is associated

[6]This discussion is only to gain insight about the meaning of the split gain, and there is no attempt to be rigorous.

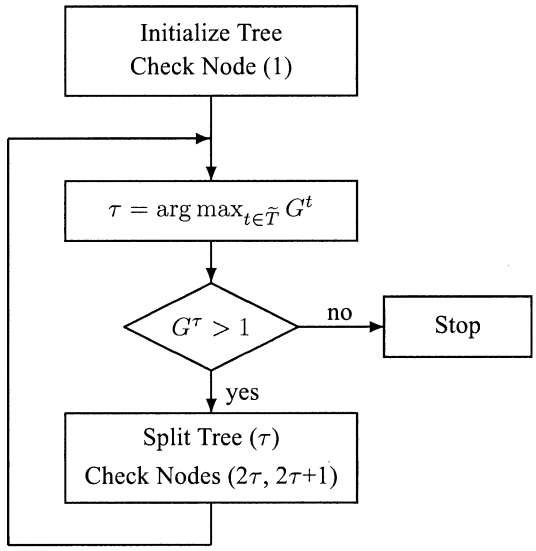

Fig. 5. High-level flow chart of the segmentation algorithm.

to it ($\mathcal{S}^1 = \mathcal{S}, y^1 = y$), and the vector of parameters $\widehat{\nu}^1$ is estimated; of course, the TS-MRF is empty ($X^T = \emptyset$).

- In the procedure CheckNode(t), the binary MRF X^t is defined on $\mathcal{S}^t$, the MAP realization $\widehat{x}^t$ is estimated together with its parameters $\widehat{\theta}^t$, and the split gain G^t is evaluated. If $G^t > 1$ this node will be split sooner or later.
- SplitTree(t) updates the structure of the tree by moving t from $\widetilde{T}$ to $\overline{T}$, and generating two new leaves $2t$ and $2t+1$; to each one of such new nodes the proper quantities ($\mathcal{S}^{2t}$, y^{2t}, $\widehat{\nu}^{2t}$, etc.) are associated (they were evaluated during the CheckNode step).

This procedure provides a fast segmentation of the image, based only on binary decisions, and solves automatically the cluster validation problem. Needless to say, this flow chart does not completely describe the segmentation algorithm because a number of details remain to be defined, such as the prior Gibbs distribution, the noise distribution, their associated parameters, the optimization technique, and the parameter estimation technique. We leave them unspecified, for the time being, because they are problem-dependent, and do not affect the overall approach. However, they influence heavily the final result, and many of the open problems that we will discuss later on concern these "details."

IV. Tree-Structured Segmentation of Remote Sensing Images

We now consider a concrete application, the segmentation of remote-sensing multispectral images, both to show the potential of the method and to highlight its weak points, to be addressed later.

A multispectral image is a collection of images of a single subject taken in different spectral windows; each component image is also called band. Color images are well-known examples of multispectral images composed of the red, blue, and green bands. Remote sensing multispectral images are composed by up to some hundreds of bands (in which case they are also called hyperspectral), so as to cover in great detail a wide spectral window and provide more valuable information about the land covers of the region under investigation [2]. The different reflection/emission properties exhibited by the materials sensed at different wavelengths represent a "spectral signature" that can be exploited to classify distinct areas of the same scene. As a matter of fact, remote-sensing multispectral images are of interest for a large number of applications, like meteorology, geology, earth resource management, pollution monitoring, not to speak of military surveillance, and most of these applications involve automatic and/or operator-assisted image segmentation and classification.

A. Statistical Model

We model a B-band multispectral image as a random field of labels X, where the vector-valued observations Y (one component for each band) obey relation (1), made more explicit here for convenience

$$Y_s = g(X_s) + N_s(X_s) \tag{17}$$

where $g(\cdot)$ is a deterministic vector-valued function of the label which accounts for the average properties of the class. The noise N is modeled as a field of zero-mean independent Gaussian random vectors whose spectral covariance matrix depends in general on the class. Therefore, the observations are conditionally independent given the label field

$$p(y|x) = \prod_s p(y_s|x_s) \tag{18}$$

with density

$$p(y_s|x_s) = \frac{1}{(2\pi)^{B/2}|\Sigma_{x_s}|^{1/2}} \cdot \exp\left[-\frac{1}{2}(y_s - g(x_s))^T \Sigma_{x_s}^{-1}(y_s - g(x_s))\right] \tag{19}$$

where Σ_k is the covariance matrix of the kth class.

The label field is modeled as a tree-structured MRF. For each component binary field we consider the same simple model, with the second-order neighborhood system, where only two-site cliques can possibly assume a non zero value. More precisely

$$V_c^t(x) = \begin{cases} \beta^t, & \text{if } x_s \neq x_r, s, r \in c \\ 0, & \text{otherwise} \end{cases}. \tag{20}$$

Single-site cliques are not used because there is no reason to favor a label over another, and larger cliques are neglected to speed-up processing. With this model, the vector of parameters θ^t associated with node t reduces to a single parameter $\beta^t > 0$, interpreted as an "edge-penalty". In fact, when $\beta^t = 0$ all realizations are equally likely, whereas larger values of β^t tend to penalize nonhomogeneous cliques making smoother realizations more and more likely. Of course, β^t is not known *a priori*, and must be estimated together with x^t.

B. Estimation

The estimates of x^t and β^t are obtained by the alternating marginal optimization, described more precisely here for the problem at hand. For the sake of notational simplicity, we drop the superscript t in the following, this should not cause ambiguities.

The initialization step is a simple clustering of the spectral vectors by the K-means algorithm [22]. Given this initial segmentation, it is straightforward to estimate the class parameters ν (means and covariance matrices) through sample means, just as is done in following steps after MRF-based segmentations. As for β, a true ML estimate is too complex, and we resort to the much simpler maximum pseudo-likelihood (MPL) estimate

$$\widehat{\beta} = \arg\max_{\beta} \prod_{s \in \mathcal{S}} p(\widehat{x}_s | \widehat{x}_{\eta(s)}, \beta) \tag{21}$$

proposed in [6]. Given a $\widehat{\beta} \neq 0$, the realization $\widehat{x}$ is now estimated by means of the ICM algorithm, which is much faster than simulated annealing but guarantees convergence only to a local optimum, depending on the initial conditions. This alternating estimation procedure goes on until convergence is reached, typically after a few iterations.

At this point, the split gain should be evaluated according to (15), but this requires computation of the partition function $Z(\widehat{\beta})$ appearing in $p(\widehat{x})$, which is prohibitive in practice. Therefore, $Z(\widehat{\beta})$ is assumed equal to the partition function of a 2-D torus having the same size as the region under analysis, for which numerical approximations are possible.

C. Numerical Experiments and Comments

To gain insight about the performance of TS-MRF segmentation we experiment with a real-world multispectral image, acquired by the GER (Geophysical Environmental Research) airborne sensor, which portrays an agricultural area in Germany near the river Rhein. The image is composed by 63 bands of 1920×512 pixels quantized at 16 bit per pixel (bpp), but here we use only six bands and a small region of 256×256 pixels.

Fig. 6 provides some visual results. In part (a), a sample band of the original test image is reported to allow for a visual inspection of results. Part (b) shows the output of a reference segmentation algorithm with $K = 5$ classes. To establish a meaningful comparison, the reference technique is equal under all respects to the proposed technique with the only difference that a flat (non structured) MRF model is used. Then, parts (c)-(f) show the output of the proposed technique at various steps, namely for K going from 2 to 5, with region colors selected so as to simplify visual analysis. Focusing on the case $K = 5$, it can be observed that both algorithms provide reasonable segmentations, which are quite similar in general, although the flat-MRF technique seems to be more accurate[7] in a few locations of the image. Both segmentation maps, however, show the same kind of problems, namely, boundary fragmentation and false contours, that will be discussed and addressed later.

[7]No ground truth is available for this image, and we can only rely on visual inspection for the time being.

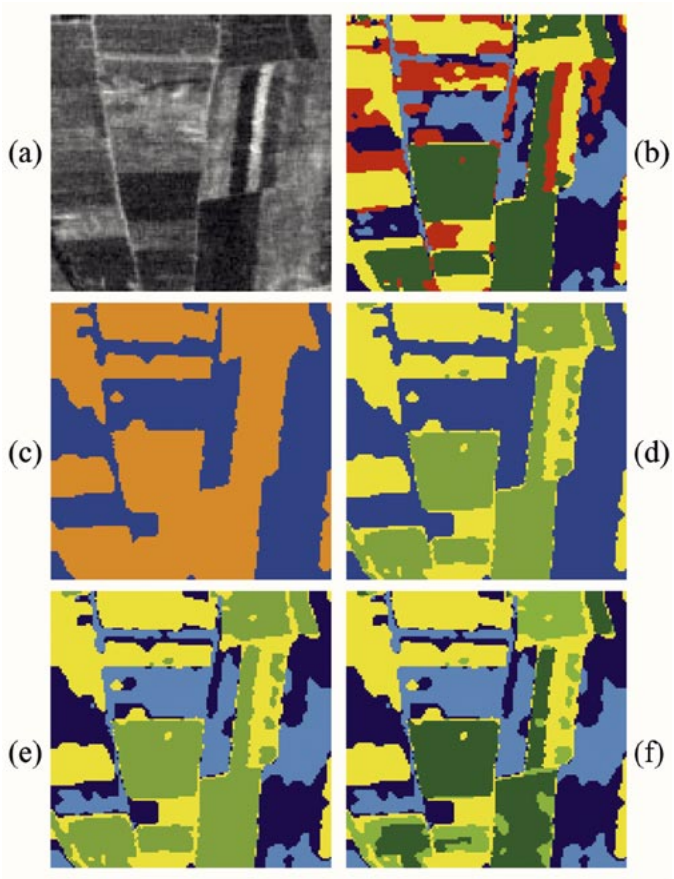

Fig. 6. Segmentation of a multispectral image: (a) one band of the original image; (b) segmentation map obtained with the reference (flat-MRF) technique; (c)-(f) 2-class to 5-class maps obtained with the proposed (TS-MRF) technique.

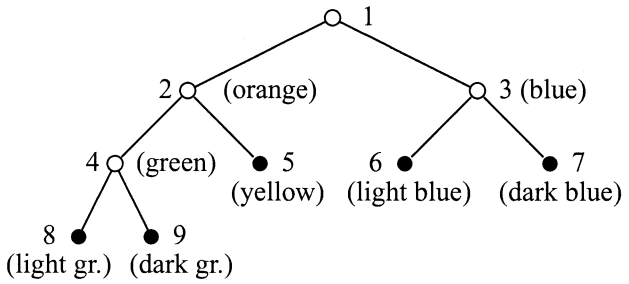

Fig. 7. Segmentation tree for the 5-class map of Fig. 6(f).

To follow the progress of the TS-MRF algorithm, in Fig. 7 we show the corresponding tree, and in Table I the most relevant parameters associated with each node, namely the estimated (log) split gain and edge penalty. Initially, the split gains are very large, indicating the coexistence in the same image of wildly different regions, while they tend to decrease when the nodes are associated with more homogeneous segments (such behavior, however, is nonmonotonous, as can be expected with real-world data). It can be observed that they are all positive, both because of noise and the presence of unconnected subregions in each class. Therefore, the algorithm would go on to deliver a larger tree with much smaller regions, it was pruned as shown in Fig. 7 only for the sake of clarity.

TABLE I
RELEVANT PARAMETERS ASSOCIATED WITH THE TREE OF FIG. 7

node	β	$\log G$
1	1.81	14690
2	0.93	3455
3	0.91	7246
4	0.86	1995
5	0.77	1214
6	1.61	2961
7	0.93	3216
8	2.54	544
9	0.94	1311

V. IMPROVED TS-MRF SEGMENTATION

By comparing the TS-MRF segmentation map with the original pseudo-color image a number of segmentation "errors" (according to a subjective judgement) clearly appear. We recognize two different types of problems in Fig. 6(f): false contours, by which we mean the unneeded segmentation of a region, originated by noise and/or by slowly varying image statistics, and boundary fragmentation, which indicates an abnormal proliferation of different labels close to the boundaries between two regions. In part, these errors can be attributed to the many simplifying choices made in the modeling and estimation steps, starting from the use of a simple model, up to the adoption of the fast but suboptimal ICM algorithm and, in fact, they also appear when a flat-MRF model is used. However, they are also a consequence of the constrained structure of the label field, and as such they will be analyzed in the following in order to obtain a more reliable segmentation algorithm.

A. Region Merging and Contour Reshaping

It is not uncommon, when dealing with real-world noisy images, that a single object happens to be over-segmented in two or more regions. This can be due to the presence of particular noise patterns, or even just because of slow spatial variations in the original data themselves. With a TS-MRF model, however, this phenomenon becomes more frequent because of the obvious inability of the algorithm to deal with nonbinary structures. The example of Fig. 8 illustrates this problem: in (a) we have a synthetic image with three distinct regions; because only binary segmentations are possible, it can easily happen (given the region geometry and the presence of noise) that the central region y is split in half among the two newly formed segments, 2 and 3 (b); after two more splits, we have the situation shown in (c), where two different segments, 5 and 6, correspond to two adjacent parts of the same region y, a clear failure of the algorithm.

Various solutions can be envisaged for this problem, such as, for example, allowing a variable number of children for each node. For the sake of simplicity and speed we decided to keep the binary structure, and to allow for the reshaping of region boundaries by means of a split-and-merge procedure [29]. To gain insight about the rationale of this approach, consider again Fig. 8: after the split of node 2, we have leaves (and regions)

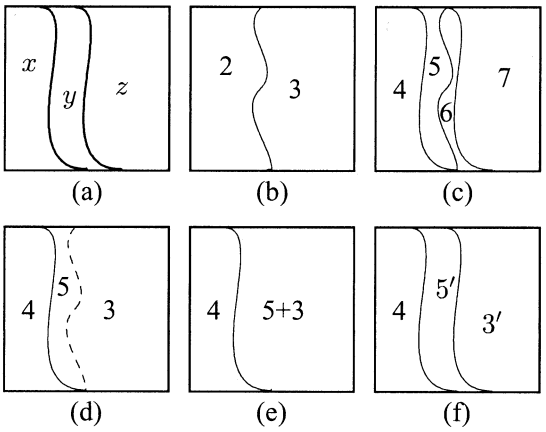

Fig. 8. Using a split-only approach, the synthetic image (a) can be erroneously segmented in four classes (c). Allowing for region merging, the correct solution is obtained (f).

4, 5, and 3, as shown in (d); at this point, we can try to merge each of the new nodes (4 and 5) with all existing leaves (only node 3 in this case) and assess the possible gain: the merging of nodes 5 and 3 should indeed provide a gain, because region y will then belong to a single class. In this case the merging takes place (e); this merged region (5+3) will eventually be split again originating regions 5' and 3', and hence the correct segmentation shown in part (f).

In Section IV, we have explained how the growth of the tree is guided by the split gain G^t which indicates whether the region $\mathcal{S}^t$ can be considered homogeneous or else should be further segmented according to the realization x^t of the associated binary MRF. Now we use the same mathematical setting to decide about a possible merging. Given the regions $\mathcal{S}^{t'}$ and $\mathcal{S}^{t''}$ associated with the leaves t' and t'', we must decide whether they should remain distinct or else be merged to form a new single region associated to node t, which provides a better statistical description. To this end, given node t, the fictitious parent of t' and t'' associated with $\mathcal{S}^t = \mathcal{S}^{t'} \bigcup \mathcal{S}^{t''}$, and given $\widetilde{x}^t$, the realization of X^t that corresponds to the current segmentation of $\mathcal{S}^t$ in $\mathcal{S}^{t'}$ and $\mathcal{S}^{t''}$, we must evaluate the quantity

$$M^t = \frac{p(y^t|\nu^t)}{p(y^t|\widetilde{x}^t)p(\widetilde{x}^t)} \tag{22}$$

which we call merging gain of nodes t' and t''. If $M^t > 1$ then the two leaves should be merged in node t, because $\mathcal{S}^t$ is better described as a single region than as the two segments currently available.

We point out that this merging operation requires a redefinition of the structure of the tree.[8] In addition, a certain number of higher level nodes, the ancestors of those involved in the merging, change their associated attributes ($\mathcal{S}^t$ and related) because a whole region switches sides. In particular, the MRF realizations implied by such a redefinition are not optimal anymore (assuming they were before).

To reduce these restructuring problems we impose an additional constraint, and require that each merging step (if ac-

[8]This also suggests a completely different approach, where the tree is built bottom up, as proposed in [21], but starting from an over-segmented image and a suitable RAG-based MRF.

cepted) must be immediately followed by a splitting step on the same node, so that the structure of the tree is not modified. The combined effect of merging and splitting is a redefinition of the boundary between the two regions associated to nodes t' and t'', which cannot but improve (by definition) the likelihood of the segmentation. This also suggests that it can be convenient to perform a merging-and-splitting operation even for nodes whose merging gain is less than one, provided that

$$M^t G^t = \frac{p(y^t|\widehat{x}^t)p(\widehat{x}^t)}{p(y^t|\widetilde{x}^t)p(\widetilde{x}^t)} > 1. \tag{23}$$

Some comments are in order here about the complexity of this modified segmentation algorithm. In fact, even though merging is a low-cost process because the MRF realization is already available (no maximization required) many new tentative splits are also necessary, and therefore the computational burden certainly increases. The number of splits that are actually tested, however, can be significantly reduced through simple "admission criteria" (e.g., do not test the split if $M^t \ll 1$), not discussed here for brevity. In addition, such splits operate on smaller and smaller regions as the procedure goes on. Therefore, the overall complexity increases with respect to the basic algorithm but remains much inferior than that of algorithms based on a flat MRF model, as will be seen in the experiments.

B. Boundary-Preserving Adaptive MRF Model

By looking at the maps of Fig. 6(f), and also Fig. 6(b), it clearly appears that several region boundaries are "fragmented," namely, there are many boundary pixels which are associated with none of the adjacent regions but have a different label attached. By visual inspection, it is clear that quite often this phenomenon does not correspond to a ground truth, namely, there is no actual "third" region in between the two neighboring regions. Instead, it is likely due to the finite resolution of the sensor which, in boundary cells, happens to integrate contributions from several land covers, originating a spectral response that is quite different from those of the adjacent regions. Therefore, in boundary cells, observed data are not very reliable.

In a binary split, the pixel is attributed to one of the two adjacent regions, but when this region is further split chances are that this "uncertain" pixel is erroneously classified. Fig. 9 clarifies this point by showing how fragmentation arises in a small subregion of Fig. 6(a). The first split separates the bright region on the left from the dark region on the right: some boundary pixels with intermediate characteristics are classified as dark. The second split operates on the dark region and further divides it: at this point, because of their mixed nature, some of the former boundary pixels are now attributed to the wrong subregion. Obviously, this makes much less sense than choosing either one of the neighboring regions and should be avoided unless the data give a clear indication. In the example selected, the segmentation map of Fig. 6(f) shows that along the boundary between the two regions (light blue and dark green) there are pixels wrongly classified as belonging to all other classes.

It is clear that the data by their very nature tend to cause such wrong segmentations near region boundaries, and in fact this phenomenon arises just as well when flat MRF models are used [see Fig. 6(b)]. In addition, prior information does not help correctly resolving such ties, because the mixed neighborhood gives fuzzy indications. However, when a tree-structured segmentation is used, a simple fix is available to make better use of contextual information [30]. Consider the example of Fig. 10(a): here the context does not give much help, because the neighborhood of the target pixel is almost evenly divided (5 "white," 3 "black") and the observed data *should* play a relevant role. In the case of Fig. 10(b), however, the target pixel is surrounded by either white or "don't care" pixels (namely, pixels outside the region of interest). Here, the relevant context is all "white" and it seems reasonable to favor strongly this hypothesis increasing the relative importance of prior information against the data. Based on these observations, we decided to change slightly the binary MRF model in order to emphasize the relative importance of the context even when the number of relevant cliques is reduced.

In our original binary MRF model the ratio of the two *a priori* probabilities given the neighborhood depended only on the number of surrounding "black" and "white" pixels, N_B and N_W, namely,

$$\frac{\Pr(X_s = W|x_{\eta(s)})}{\Pr(X_s = B|x_{\eta(s)})} = \exp[\beta(N_W - N_B)]. \tag{24}$$

With this definition, a 5–3 and a 2–0 context are equivalent. To restore the value of prior information, we simply increase adaptively the edge penalty value when the context is not complete.

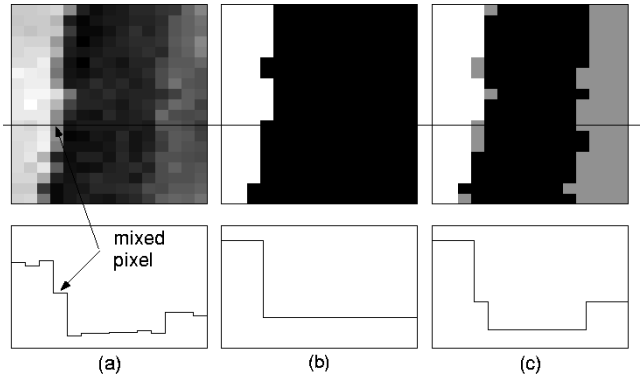

Fig. 9. Boundary pixels between two regions often have mixed features (a); they can belong to either one of the neighboring regions (b), but in subsequent splits can be erroneously associated with a "third" class (c).

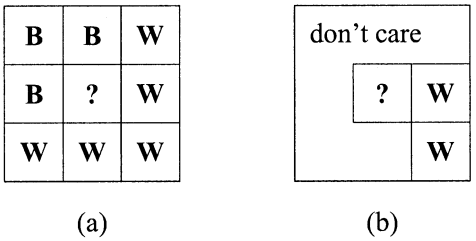

Fig. 10. Context is mixed in (a) and does not help making a decision, but it is very biased in (b) and this should be properly taken into account.

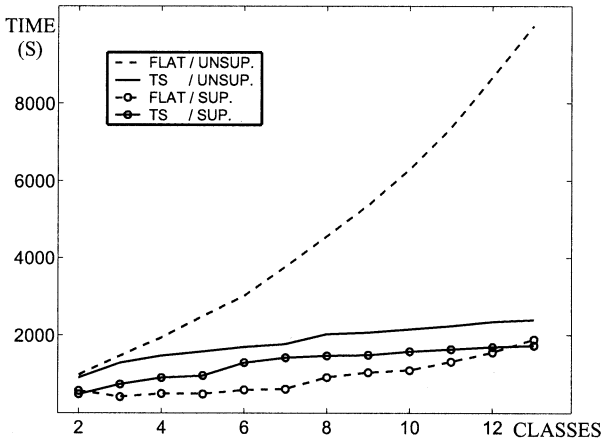

Fig. 11. CPU time as a function of the number of classes, for "flat" and "TS" algorithms. In the unsupervised case, the TS algorithm is clearly preferable.

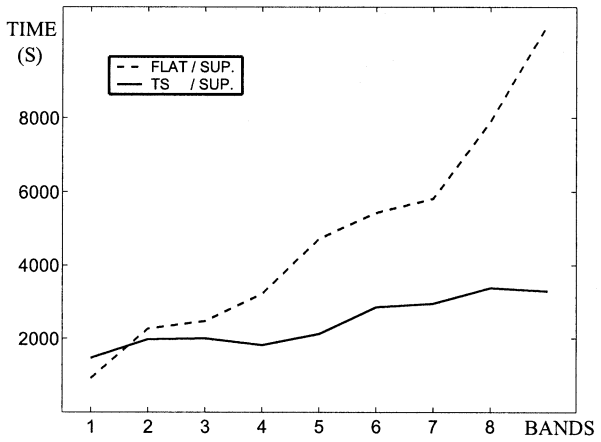

Fig. 12. CPU time as a function of the number of bands, for "flat" and "TS" algorithms. The TS algorithm is more and more convenient as the number of bands increases.

In particular, we stretch the range of values to its original width by setting

$$\beta' = \frac{8}{N_B + N_W}\beta \tag{25}$$

so that a 2–0 context, for example, becomes equivalent to an 8–0 one. Of course, other definitions can be used instead of (25), but this simple modification has provided very good results in the experiments.

VI. Experimental Results

In this section, we try to assess the performance of the proposed segmentation procedure by means of some numerical experiments on real-world and synthetic multispectral images.

A. Computational Complexity

First of all, we want to substantiate our initial claim about the speed of TS-MRF-based segmentation. To this end we consider again the GER image described above (a 512×512 pixel section) and segment it using both the proposed and reference algorithms which, from now on, will be referred to simply as "TS" and "flat" algorithms.

Fig. 11 reports CPU time (on a Pentium-II 533 MHz machine) as a function of the number of classes when a single-band image is considered. Contrary to our intuition, in the supervised case, when the number of classes is known *a priori*, the flat algorithm is almost always faster than the TS one. As a matter of fact, the tree structure entails the additional burden of dealing with irregular lattices, which calls for more complex data structures and additional controls. On the other hand, the simple MRF model used in both cases, together with the suboptimal ICM algorithm which guarantees a quick convergence, reduce the potential computation gain of using only binary fields. Such an advantage could become significant with more sophisticated MRF models and optimization algorithms.

However, if we consider the unsupervised case, the picture changes completely. With the flat algorithm, to obtain and accept a K-class map, all independent segmentations with $k = 2, \ldots, K+1$ classes are needed in order to evaluate their validity measure. With the TS algorithm, instead, all segmentations are nested, and the tests required to stop the growth are all local: therefore, given the K-class map, only another layer of splits is required, one for each leaf of the tree, with a nearly constant additional complexity with respect to the supervised case. As a consequence, using the TS-MRF model becomes more and more convenient as the number of classes grows (which is often the case for large images).

This advantage increases when we consider multiple bands. Fig. 12 reports the CPU time as a function of the number of bands in the image for both the flat and TS algorithms when eight classes are considered (supervised case). Complexity grows quite fast for the flat algorithm, and it soon exceeds that of the TS algorithm, which in turn remains almost constant. The effects of multiple bands on the complexity of the *unsupervised* case can be easily extrapolated without further experiments.

B. Segmentation Accuracy

Let us now study the reliability of the proposed segmentation technique. To this end we build a synthetic image starting from a 256×256 pixel section of a 6-band Thematic Mapper image of a region near Lisbon, Portugal. The original image is segmented using a flat-MRF model with 5 classes, and the result is accepted as our ground truth, shown in Fig. 13(a). In the synthetic image, each (vector-valued) pixel is replaced by the spectral signature of the corresponding class. To simulate the behavior of a limited-resolution sensor, the image is low-pass filtered and, finally, white Gaussian noise is added. The result is shown in Fig. 14(a).

In Fig. 13, together with the ground-truth, we show the segmentation maps produced by various algorithms, that is, vector quantization (b), flat-MRF (c), TS-MRF (d), TS-MRF with adaptive model (e), and TS-MRF with adaptive model and merging (f). All algorithms correctly stop at $K = 5$ (this was an input parameter for VQ), which is reasonable for such a simple image. The misclassification rates are 10.5%, 5.4%, 9.6%, 9.4%, and 5.5%, respectively. These numbers, however, are not sufficient to describe the different behaviors and a deeper analysis is required.

Vector quantization (minimum distance clustering) for example has almost the same error rate as other techniques, but

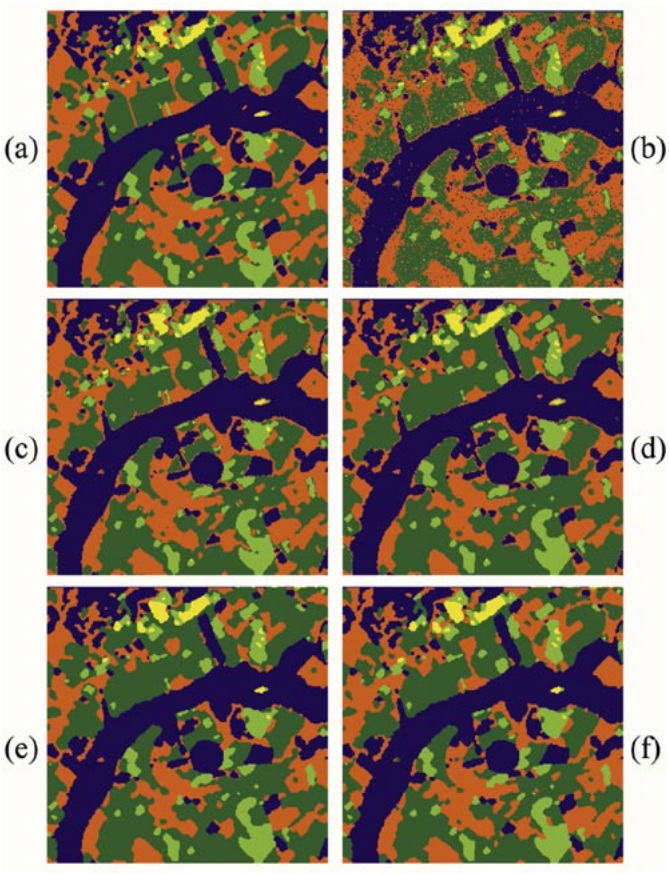

Fig. 13. Synthetic multispectral image: ground truth (a), and segmentation maps produced by VQ (b), flat-MRF (c), TS-MRF (d), adaptive TS-MRF (e), adaptive TS-MRF with merging (f).

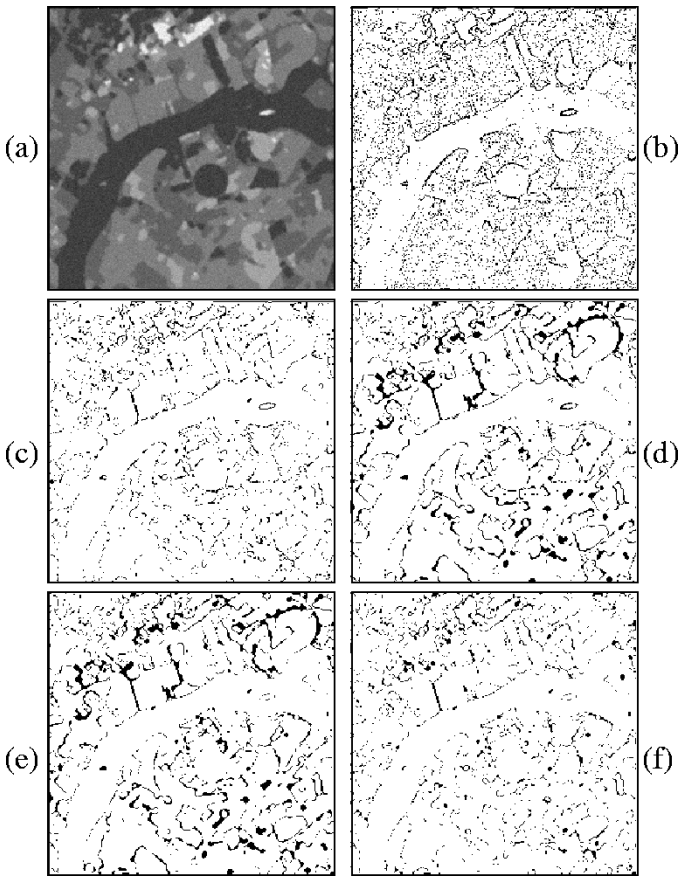

Fig. 14. Synthetic multispectral image: one band of the original image (a), and classification errors with VQ (b), flat-MRF (c), TS-MRF (d), adaptive TS-MRF (e), adaptive TS-MRF with merging (f).

the corresponding classification map is very poor, with a large number of isolated errors [see also the error map in Fig. 14(b)]. This is particularly annoying when the map is the input for further automated processing steps, such as shape extraction, or data compression, etc, whose performance will significantly deteriorate with such a "noisy" map. The TS-MRF algorithm provides a much smoother map, which is desirable, however it still presents a large number of errors, especially in boundary regions. In addition, although isolated errors have disappeared, entire (small) regions are now wrongly classified, as clearly appears by the error map reported in Fig. 14(d). Under this respect, VQ might even be preferable. The introduction of a boundary-adaptive MRF model does not reduce much the error rate, and relatively large regions keep being misclassified [see Fig. 14(e)]. However, the boundary fragmentation phenomenon is now absent [Fig. 13(e)]: many boundary pixels are erroneously classified, because of the joint effects of filtering and additive noise, but they are always attributed to one of the neighboring regions, leading to a smoother and more acceptable segmentation. Finally, using both the adaptive model and the split-and-merge procedure, a segmentation map is obtained which closely resembles that of the flat-MRF algorithm (Figs. 13(c), (f) and 14(c), (f)), with an error rate that now is just above 5%. The loss of entire regions is almost absent and, in any case, it is not imputable to the tree structure, as it occurs just as well when the flat algorithm is used, but rather to the overly simple MRF model.

In summary, with the proposed improvements, the tree-structured model seems just as able as the flat model to faithfully catch the image structure, with the side benefits of reducing computation time, and providing a whole tree of intermediate maps and associated parameters to help image analysis.

It should be noted that the image considered above was not particularly easy to analyze with a tree-structured approach. The power of the TS-MRF model becomes clear, instead, when the image exhibits an inherent hierarchical structure. To analyze this situation, let us begin by considering a single-band synthetic image built by repeated use of a Gibbs sampler, first on the whole set of sites $\mathcal{S}^1$ with $\beta^1 = 1$, and then on $\mathcal{S}^2$ with $\beta^2 = 0.3$; see Fig. 15(a). The addition of white noise leads to the image shown in Fig. 15(b), having a signal-to-noise ratio (SNR) of 14 dB. In Fig. 15 part (c) and (d) we show the result of segmentation carried out by flat and TS algorithms, respectively. The difference is striking: while the TS algorithm faithfully catches all details of the ground truth, with an overall misclassification rate of 5.2%, the flat algorithm performs a brutal smoothing of the boundaries, with a much increased misclassification rate, 14.3%, and a clear (even visually) failure to correctly model the image. This is not surprising, because the flat algorithm estimates a single edge penalty for the whole image, $\widehat{\beta} = 0.853$, which lies in between the two correct values, and is too large to correctly model the fine-grain part of the image. On the contrary, the TS algorithm estimates only the first edge

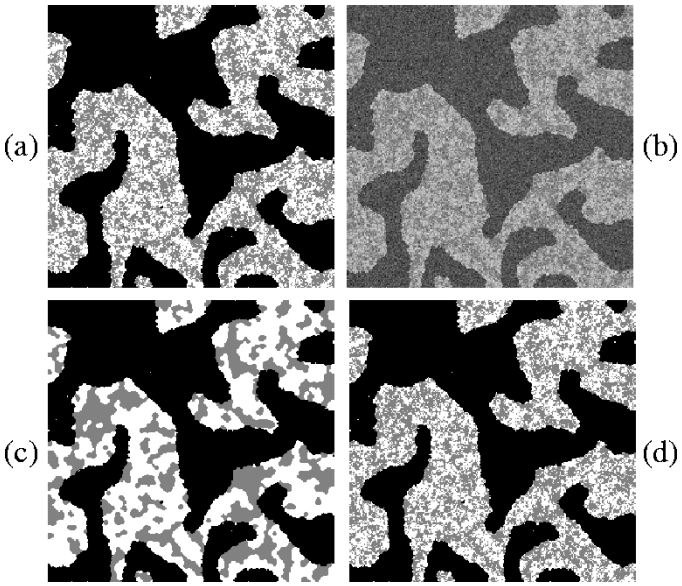

Fig. 15. Synthetic image produced by a Gibbs sampler: ground truth (a), original noisy image (b), and segmentation maps produced by the reference (c) and proposed algorithm (d).

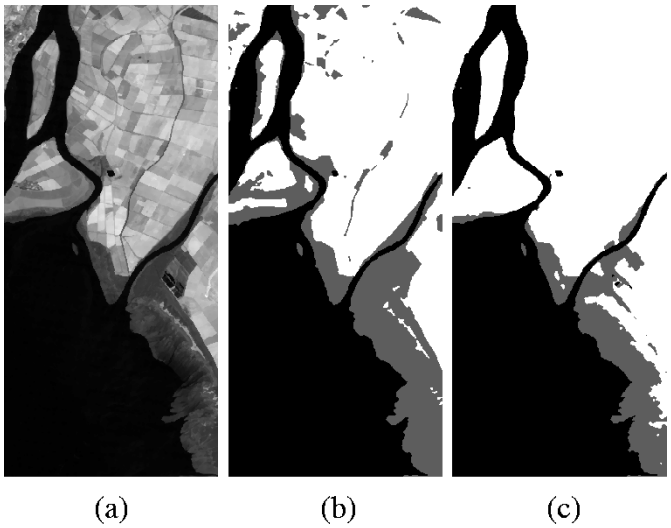

Fig. 16. Land-and-ocean real-world multispectral image: one band of the original image (a), and 3-class segmentation maps produced by the reference (b) and proposed algorithm (c).

penalty on the whole image, obtaining $\widehat{\beta}^1 = 1.094$, and uses it to carry out accurately the first 2-class segmentation; then it estimates the second edge penalty only on class-2 pixels, obtaining again an accurate value, $\widehat{\beta}^2 = 0.301$, and a precise segmentation of the region of interest.

Even though the former synthetic image was obviously built so as to fit a tree-structured model, it is not difficult to figure out real situations, where this model is appropriate. The image considered in next experiment, a six-band 256×512 pixel multispectral image taken by the Landsat TM sensor, could be one such example, since the statistics of the ocean region are likely to be quite different from those of the rest of the image [see Fig. 16(a)]. Although a performance analysis is not feasible in this case, the behavior of the TS algorithm merits some comments. In the first step, it separates the ocean region from the rest of the image, and then, operating only on the land region, is able to single out a definite "coast/river banks" class, before going on with further segmentation [see Fig. 16(c)]. The flat algorithm, instead, when requested to segment the image in three classes, succeeds in isolating the ocean region, but then clusters regions of different nature in both remaining classes providing results that are not so easily interpreted.

Fig. 17. One band of a real-world multispectral image.

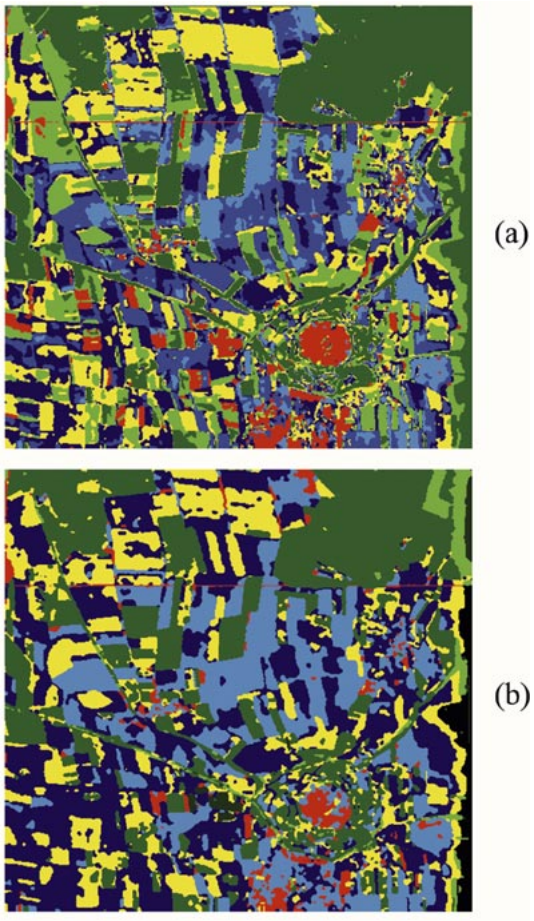

Fig. 18. Segmentation maps for the multispectral image of Fig. 17 produced by the reference (a) and proposed algorithm (b).

To conclude this section, let us consider the seven-class segmentation of a real-world multispectral image, a 512×512

pixel section (Fig. 17) of the 8-band GER image considered above. Results are shown in Fig. 18. For such a noisy, and relatively large image, segmentation is quite a challenging task and in fact the flat and TS algorithms provide markedly different maps. In both cases, most fields are clearly recognizable, together with the artificial structures (roads and a city), and woods. A major difference is that the TS algorithm separates the region on the right edge of the image (black in the map), clearly affected by some sort of acquisition disturbance, from the adjacent woods and fields. The flat map seems to better preserve details but it also suffers from boundary fragmentation which is absent in the TS map. Without the help of an external reference, it is not easy to understand which map better represents the situation, but it seems safe to say that, even for this apparently flat-structured image, tree-structured segmentation does not seem to impair quality.

VII. Conclusions and Future Research

We have presented a new segmentation algorithm based on a tree-structured MRF model. The algorithm can be quite fast (also depending on important implementation details, such as the binary-MRF model, the estimation techniques, etc.) because all segmentations are binary, and they operate on smaller and smaller regions as the tree grows; in addition, such segmentations are completely independent of one another, allowing for node-parallel and recursive implementation. The cluster validation problem is solved in a seamless way and no extra processing is required. All parameters are estimated locally, allowing for spatial adaptivity, and large-scale and fine-scale descriptions can be decoupled. Texture, for example, can emerge down the tree, and be different from region to region. The output of the algorithm is not just the segmentation, but the whole tree, with all parameters associated with the individual nodes. Hence, given a K-leave segmentation tree, all intermediate trees and associated segmentation maps can be obtained by pruning back nodes in reverse order.

Numerical experiments show that the proposed algorithm performs very well on simple, relatively clean, images, while it exhibits a more controversial behavior on large noisy images. In any case, the performance is very close or superior (depending on the image) to that of a reference unconstrained algorithm based on a "flat" MRF model.

We feel that the TS-MRF model can be a powerful tool for image segmentation, and work is under way to better understand its potential and to implement faster and more reliable algorithms.

Given the overall structure of the algorithm, a few points certainly deserve greater attention, that is, the binary MRF model, the optimization technique, and the parameter estimation technique. In this work, we have consistently privileged simplicity, but a more complex MRF model, with larger cliques and more parameters, would certainly allow for a better segmentation at each stage of the process. The introduction of a simple adaptive model, for example, has already led to significant improvements. Second, a more reliable optimization technique should be used in place of the ICM, to reduce problems related to suboptimal binary segmentations. To avoid slowing down the algorithm too much, a multi-resolution approach, such as [9] or similar, could be easily embedded in our hierarchical model. Finally, more attention should be devoted to the estimation of important parameters, first of all the edge penalties, and techniques like EM and ICE should be tested and finally introduced in the algorithm.

In parallel to these tasks, we want to consider a different structure for the algorithm, based on the over-segmentation of the image followed by the systematic merging of elementary regions. Experience with classification trees [17] shows that a large performance gain is possible when a bottom-up approach is adopted. This alternative strategy could draw ideas and methods both from non-MRF binary description trees [21], and from RAG-based flat-MRF techniques.

Finally, we want to better explore the applications of the proposed segmentation technique. A first goal is to develop and assess an image compression scheme based on TS-MRF segmentation, improving on what already proposed in [26]. In addition, the use of TS-MRF segmentation as an interactive tool to help in image classification and analysis is also worth exploring.

References

[1] A. Richards, *Remote Sensing Digital Analysis: An Introduction*, 2nd ed. Berlin, Germany: Springer Verlag, 1993.

[2] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Processing Mag.*, vol. 19, pp. 17–28, Jan. 2002.

[3] L. Torres and M. Kunt, *Video Coding: The Second Generation Approach*. Norwell, MA: Kluwer, 1996.

[4] J. Besag, "Spatial interaction and the statistical analysis of lattice systems," *J. R. Statist. Soc.*, ser. B 36, pp. 192–236, 1974.

[5] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, pp. 721–741, Nov. 1984.

[6] J. Besag, "On the statistical analysis of dirty pictures," *J. R. Statist. Soc.*, ser. B 48, pp. 259–302, 1986.

[7] G. Winkler, *Image Analysis, Random Fields and Dynamic Monte Carlo Methods*: Springer-Verlag, 1995.

[8] S. Z. Li, *Markov Random Field Modeling in Image Analysis*: Springer-Verlag, 2001.

[9] C. A. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Transactions on Image Processing*, pp. 162–177, Mar. 1994.

[10] Z. Kato, M. Berthod, and J. Zerubia, "A hierarchical Markov random field model and multi-temperature annealing for parallel image classification," *Graphical Models and Image Processing*, vol. 58, pp. 18–37, Jan. 1996.

[11] J. M. Laferté, P. Pérez, and F. Heitz, "Discrete Markov image modeling and inference on the quadtree," *IEEE Trans. Image Processing*, pp. 390–404, Mar. 2000.

[12] M. Mignotte, C. Collet, P. Pérez, and P. Bouthemy, "Sonar image segmentation using an unsupervised hierarchical MRF model," *IEEE Trans. Image Processing*, pp. 1216–1231, July 2000.

[13] I. Y. Kim and H. S. Yang, "A systematic way for region-based image segmentation based on Markov random field model," *Pattern Recognit. Lett.*, pp. 969–976, Oct. 1994.

[14] A. Sarkar, M. K. Biswas, B. Kartikeyan, V. Kumar, K. L. Majumder, and D. K. Pal, "A MRF model-based segmentation approach to classification for multispectral imagery," *IEEE Trans. Geosci. Remote Sensing*, pp. 1102–1113, May 2002.

[15] Y. Tsaig and A. Averbuch, "Automatic segmentation of moving objects in video sequences: A region labeling approach," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 597–612, July 2002.

[16] G. Poggi and A. R. P. Ragozini, "Image segmentation by tree-structured Markov random field," *IEEE Signal Processing Lett.*, pp. 155–157, July 1999.

[17] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Belmont, CA: Wadsworth, 1984.

[18] S. Z. Li, "On discontinuity-adaptive smoothness priors in computer vision," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 576–586, June 1995.

[19] P. C. Smits and S. G. Dellepiane, "Discontinuity-adaptive Markov random field model for the segmentation of intensity SAR images," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 627–631, Jan. 1999.

[20] P. Salembier, F. Marques, M. Pardas, J. R. Morros, I. Corset, S. Jeannin, L. Bouchard, F. Meyer, and B. Marcotegui, "Segmentation-based video coding system allowing the manipulation of objects," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 60–74, Feb. 1997.
[21] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval," *IEEE Trans. Image Processing*, pp. 561–576, Apr. 2000.
[22] R. O. Duda, D. G. Stork, and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 2000.
[23] A. D. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc.*, ser. B 39, pp. 1–37, 1977.
[24] F. Salzenstein and W. Pieczynski, "Global and local methods of unsupervised Bayesian segmentation of images," *Graph. Models Image Process.*, vol. 59, pp. 205–220, 1997.
[25] A. Mohammad-Djafari, "Joint estimation of parameters and hyperparameters in a Bayesian approach of solving inverse problems," *Proc. 1996 IEEE Int. Conf. Image Processing*, vol. 2, pp. 473–476, 1996.
[26] G. Gelli and G. Poggi, "Compression of multispectral images by spectral classification and transform coding," *IEEE Trans. Image Processing*, pp. 476–489, Apr. 1999.
[27] K. L. Oehler and R. M. Gray, "Combining image compression and classification using vector quantization," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 461–473, 1995.
[28] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE Trans. Inform. Theory*, vol. 35, pp. 299–315, Mar. 1989.
[29] C. D'Elia, G. Poggi, and G. Scarpa, "Advances in the segmentation and compression of multispectral images," in *Proc. 2001 IEEE Int. Geoscience Remote Sensing Symp.*, vol. 6, July 2001, pp. 2671–73.
[30] ——, "An adaptive MRF model for boundary-preserving segmentation of multispectral images," in *Proc. XI Eur. Signal Processing Conference*, Toulouse, France, Sept. 2002.

Ciro D'Elia was born in Naples, Italy, in 1971. He received the Dr. Eng. degree in telecommunication engineering in 1998, and the Ph.D. degree in computer science and electronic engineering in 2001, both from the University Federico II, Naples.

Since October 2001, he has been a Researcher with the University of Cassino, Italy. His research interests are in statistical signal processing (in particular wavelets and data compression) and telecommunications networks.

Giovanni Poggi was born in Naples, Italy, on February 17, 1963. He received the Dr. Eng. degree in electronic engineering from the University Federico II, Naples, in July 1988.

From 1990 to 1998 he was a Researcher with the Department of Electronic and Telecommunication Engineering, University Federico II, and in the same University has been an Associate (from 1998 to 2002) and a Full (since November 2002) Professor of Telecommunications. In 1992, he was a Visiting Scholar in the Department of Electrical Engineering, Stanford University, Stanford, CA. His current research activity is in statistical signal processing, and in particular compression and segmentation of remote-sensing images.

Giuseppe Scarpa was born in Pompei, Italy, on January 5, 1975. He received the Dr. Eng. degree in telecommunication engineering from the University Federico II, Naples, Italy, in March 2001. Since November 2001, he has been pursuing the Ph.D. degree in the Department of Electronic and Telecommunication Engineering, University Federico II.

His current research lies in the field of statistical image processing.