# Inference and parameter estimation on hierarchical belief networks for image segmentation

Christian Wolf[1] and Gérald Gavin[2]

**Technical Report LIRIS**

Laboratoire d'informatique en images et systèmes d'information
LIRIS UMR CNRS 5205
University of Lyon

[1]Bât. J.Verne, 20, Av. Albert Einstein
69621 Villeurbanne cedex, France
Email: christian.wolf@liris.cnrs.fr

[2]Batiment Nautibus, 8, Boulevard Niels Bohr,
69622 Villeurbanne cedex, France
Email: gerald.gavin@liris.cnrs.fr

October $21^{st}$, 2008

**Abstract**

**We introduce a new causal hierarchical belief network for image segmentation. Contrary to classical tree structured (or pyramidal) models, the factor graph of the network contains cycles. Each level of the hierarchical structure features the same number of sites as the base level and each site on a given level has several neighbors on the parent level. Compared to tree structured models, the (spatial) random process on the base level of the model is stationary which avoids known drawbacks, namely visual artifacts in the segmented image. We propose different parameterizations of the conditional probability distributions governing the transitions between the image levels. A parametric distribution depending on a single parameter allows the design of a fast inference algorithm on graph cuts, whereas for arbitrary distributions, we propose inference with loopy belief propagation. The method is evaluated on scanned document images from the $18^{th}$ century, showing an improvement of character recognition results compared to other methods. Keywords**

Markov cubes, markov models, causal hierarchical models, factor graphs, belief propagation, image segmentation, Bayesian estimation, Maximum a posteriori

## 1 Introduction

Belief networks, image segmentation, hierarchical model, graph cuts

## 2 Introduction

Image segmentation techniques aim at partitioning images into a set of non overlapping and homogeneous regions. Simple techniques, as for instance thresholding or K-means clustering, exploit information from feature space only (gray values, colors, spectral components) to classify each pixel. Probabilistic graphical models are widely used to incorporate spatial dependencies between the image pixels into the classification process, combining observed nodes and hidden nodes and their interactions. Very often Bayesian methods are employed in order to combine models of the observation process (the likelihood of the observation given a label configuration) with models on the spatial interaction (the prior knowledge). The main objective of graphical models is to model the joint probability distribution of the variables in order to be able to sample from it and to estimate the most probable configuration of hidden variables given the configuration of observed variables based on given risk functionals.

In their seminal paper [10], Geman and Geman introduced a maximum a posteriori (MAP) estimation technique for Markov Gibbs random fields (MRF) based on Gibbs sampling and simulated annealing. The sites of

the random field correspond to the pixels of the image and interactions between the sites are modeled using energy functions defined on the maximum cliques of the neighborhood graph. The Markov condition, which states that two sites are conditionally independent given realizations of their neighbors (or more precisely: $P(X_s{=}x_s|X_r{=}x_r, r \neq s) = P(X_s{=}x_s|X_r{=}x_r, r \in N_s)$, where $N_s$ is the set of neighbors of sites $s$), allows for dependence of a given site on a site further away than the local neighborhood. Although the update equations are based on the local neighborhood of each pixel, the long run behavior of the estimation algorithm takes into account these long distance interactions.

An alternative to the two-dimensional MRFs are hidden Markov chains (MC) on a Hilbert-Peano scan of an image [1]. The disadvantage of the weaker spatial interactions of the nodes in the chain is compensated by the substantially lower computational complexity achieved by non iterative Viterbi like algorihms. Hybrid models, where a Markov chain based segmentation algorithm is used to initialize an iterative MRF based segmentation, have been proposed recently [9].

Kuo and Agazzi extended the Markov chain model to a pseudo 2D graph structure, where single rows form Markov chains and the image (patch) consists of an additional Markov chain formed from super states which correspond to the individual rows [16]. The model does not share the connectivity of a full 2D MRF model but keeps the speed advantage of a Markov chain model. A full extension to a 2D connectivity as proposed by Levin et al is of exponential complexity [19].

Hierarchical models introduce a scale dependent component into the classification algorithm, which allows the algorithm to better adapt itself to the image characteristics. This is justified by the hypothesis that second order image statistics are scale dependent. Hierarchical MRFs have been introduced by Bello [2], combining a stack of flat MRFs with spatial cliques only. The solution at each scale is used as an initialization for the inference of the next finer level, which speeds up convergence for this level. In this solution, there is no real coupling between the different scales apart from the initialization. Kate et al introduce a MRF model based on a pyramidal hierarchical graph structure [12], featuring horizontal and vertical single level cliques as well as vertical cross level cliques. The scale causal multi-grid introduced by Mignotte et al. features similar inter-level cliques [20].

Bouman and Shapiro were among the first to propose causal hierarchical models [3]. In their work, a quad tree models the spatial interactions between the leaf pixel sites through their interactions with neighbors in scale. Markov chains run from a single root node to each pixel node, where all nodes apart from the leaf nodes are virtual nodes which do not correspond to an image pixel. The authors propose a sequential maximum a posteriori (SMAP) estimator, which weights misclassifications of sites on different levels differently. The quad tree shares the main advantage of causal hierarchical models, since its transition probabilities in scale can be chosen to be independent of the scale. Furthermore, the tree structure makes efficient non iterative estimation algorithms possible. The main problem of the quad tree structure is the non stationarity it induces into the random process of the leaf sites, explained by the fact that two neighboring pixels may or may not share a common parent node depending on their position on the grid. This results in visible "blocky" artifacts in the segmented image.

In the same paper [3], Bouman and Shapiro also propose a second model where each node has three parents. At first sight, the structure of the dependency graph is similar to our solution (which features four parents for each site), however, the model proposed by Bouman is a pyramidal model in that the number of nodes decreases at each level. Moreover, the inference algorithm is not the same. In both cases, the exact solution cannot be calculated. In [3], the approximation supposes a hybrid model, where, during the inference algorithm, for each site at a given level $n$, the model is supposed to be a quad tree for all levels $< n$ and a cyclic graph for all levels $> n$. The structure of the dependency graph therefore changes during the inference algorithm. In our work, the whole graph keeps its full connectivity. The cycle problem is circumvented using approximative inference through loopy belief propagation.

The graphical structure of the prism machine introduced by Rosenfeld [25] shares some similarities with our model: both graphs are not pyramidal, i.e. each level features the same number of nodes as the base level, and in both cases the connectivity spreads exponentially with increasing level. However, Rosenfelds graph has been designed for parallel computing, not inference, therefore the connectivity is not as large.

The Markov quadtree model has been refined by Laferte et al. [17]. The authors propose different estimation techniques (MAP with a Viterbi like algorithm as well as maximum of posterior marginals (MPM)) and an unsupervised parameter estimation technique adapted to the quad tree based on the expectation-maximization algorithm (EM) [6].

This first generation of Markov models suffers from several shortcomings which have been addressed recently. One of the problems is the independence of the observed variables conditional to the hidden vari-

ables and other related hypothesis usually assumed in the classical MRF and MC frameworks. In the framework of the classical generative models, relaxing this constraint by allowing a dependency of each observed node to the whole set label nodes makes inference intractable.

In a random field including level wise and inter level quad tree cliques, Wilson and Li propose a solution where the observation model is defined through the differences between neighboring sites [31]. A more general framework addressing these issues are the recently proposed conditional random fields (CRF), initially proposed with a chain structure for the labeling of sequences [18]. They tackle the problem by defining a label field which is Markovian conditioned on the observation field: $P(X_s{=}x_s|Y = y, X_r{=}x_r, r \neq s) = P(X_s{=}x_s|Y = y, X_r{=}x_r, r \in N_s)$, where $X$ is the label field and $Y$ is the observed field. In other words, instead of defining the model generating the observations from the "true" label field, the model directly expresses the posterior probability $P(X = x|Y = y)$ avoiding overly simplifying the assumptions on the true underlying generative model. Compared to classical MRFs, the clique potential functions are defined over the labellings of the maximal cliques of the label nodes $X$ *and* the full set $Y$ of observed nodes.

One of the shortcomings of classical ("generative") models is the independence hypothesis of the observed variables conditional to the hidden ones. Different solutions have been proposed [24, 31] but the most widely used are "discriminative" or conditional random fields (CRF) [18], initially proposed for chain structured graphs. The concept has been extended to graphs on a two dimensional regular lattice [15], spatio-temporal graphs [30] and arbitrary graphs[26].

The work described in this paper concentrates on the solution to the lack of shift invariance of the quad tree. We propose a new generative model, a forthcoming paper will introduce a discriminative model based on the same graphical structure. Our new model combines several advantages:

- Adaptation to the image characteristics with a hierarchical graph structure (similar to the quad tree)
- A stationary random process at the base level (where each site corresponds to one pixel of the input image).
- Fast inference using minimum cut/maximum flow algorithms for a subclass of transition probability distributions.

The paper is organized as follows: section 3 describes the quad tree structured network and section 4 extends it to the cube. Section 5 presents an inference algorithm using loopy belief propagation and section 6 outlines an interpretation of the hidden variables of the model. Section 7 presents a fast inference algorithm for a parametric class of transition probability distributions. Section 8 describes parameter estimation for the latter class of distributions and section 6 introduces an estimation technique for a nonparametric family of transition probability distributions. Section 9 discusses the computational complexity and memory requirements and section 10 experimentally validates the method. Finally, section 11 concludes.

# 3 Quad tree structured models

In the following we describe graphical models defined on a directed graph $\mathcal{G} = \{G, E\}$, where $G$ is a set of nodes (sites) and $E$ is a set of edges. The edges of the graph assign, to each site $s$, a set of parent sites (written as $s^-$) and a set of children sites (written as $s_-$). The set of all descendants of a sites $s$ is denoted $s_+$, the set of all ancestors a sites $s$ is denoted as $s^+$. $s_-, s_+, s^-$ and $s^+$ may be empty. The hierarchical nature of the graph partitions the set of nodes into levels $G^{(i)}, i \in 0..L-1$, $G^{(0)}$ being the base level corresponding to finest resolution.

Each site $s$ is assigned a discrete random variable $X_s$ taking values from the label set $\Lambda = \{0, \ldots, C-1\}$ where $C$ is the number of classes[1]. $X_G$, or short $X$ denotes the field of random variables of the graph, whereas $X_{G^{(l)}}$ denotes the field of random variables at level $l$. The space of all possible configurations of the field $X$ is denotated as $\Omega = \Lambda^{|G|}$.

The graph structure induces a specific joint probability distribution of the random field $X$:

$$P(X = x) = \prod_{s \in G} P(X_s = x_s | X_{s^-} = x_{s^-})$$

As usual, uppercase letters denote random variables or fields of random variables and lower case letters denote realizations of values of random variables or of fields of random values. In particular, $P(X = x)$ will be abbreviated as $P(x)$ when it is convenient.

[1] Unfortunately the two communities of image processing and machine learning did not coordinate their notation. As a result, in most image processing papers the symbol $X$ determines the set of hidden variables and the symbol $Y$ determines the set of observed variables, whereas in most papers in the machine learning community, especially the ones on discriminative models, the opposite has been chosen.

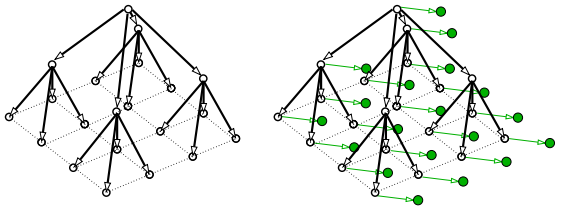

Figure 1: The Markov quad tree with (a) and without (b) observed nodes (shaded).

In the case of the Markov quad tree model [3][17], the graph $G$ forms a tree structure with a single root node $r \in G^{(L-1)}$, four children nodes for each node and a single parent node for each node except the root node (see figure 1a). Each path from the root site to one of the leaf sites forms a first order Markov chain satisfying the Markov property:

$$P(x_s|x_{G\setminus s}) = P(x_s|x_{s^-}) \quad \forall s \in G$$

The field $X$ is hidden, the objective of the inference algorithm is to estimate its values given the field of observed nodes $Y$. Each observed variable $Y_s$ is related to a hidden variable $X_s$ and is conditionally independent of the other variables given the realization of the related hidden variable:

$$\begin{aligned} P(y_s|x) &= P(y_s|x_s) \quad \forall s \in G \\ P(y|x) &= \textstyle\prod_{s\in G} P(y_s|x_s) \end{aligned} \tag{1}$$

This can be seen in the full dependency graph in shown figure 1b, where each shaded observed node is connected to its related hidden variable only.

The joint probability distribution of the full graph (including observed nodes) factorizes as follows:

$$\begin{aligned} P(x,y) &= \prod_{s\in G} p(x_s|x_{s^-}) \prod_{s\in G} p(y_s|x_s) \\ &= p(x_r) \prod_{s\in G^{(0)}\ldots G^{(L-2)}} p(x_s|x_{s^-}) \prod_{s\in G} p(y_s|x_s) \end{aligned}$$

As usual in Bayesian estimation techniques, the model can therefore be seen as a combination of a likelihood factor $P(y|x)$ and a factor describing the a priori knowledge on the estimated labels $P(x)$. The objective is to estimate the hidden variables $x$ given the observed variables $y$ given a cost functional $C(.,.)$ which determines the "punishment" of a given estimation $x'$ compared to the "real" label field $x^*$:

$$\widehat{x} = \arg\min_{x\in\Omega} E\left[C(X,y)|Y=y\right]$$

In this paper we only consider the maximum a posteriori estimation (MAP) estimation technique, which corresponds to the following cost functional:

$$C(x,x') = 1 - \delta_{x,x'}$$

where $\delta_{i,j}$ is the Kronecker delta given as

$$\delta_{i,j} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{else} \end{cases}$$

The MAP estimator is given as the mode of the posterior distribution:

$$\begin{aligned} \widehat{x} &= \arg\max_{x\in\Omega} p(x|y) \\ &= \arg\max_{x\in\Omega} \frac{p(x)p(y|x)}{p(y)} \\ &= \arg\max_{x\in\Omega} p(x)p(y|x) \end{aligned} \tag{2}$$

Other estimation techniques on the quad tree (sequential MAP and maximum of posteriori marginals) are described in [3][17].

Direct evaluation of equation (2) is intractable because of the large size of the configuration space ($|\Omega| = C^{|G|}$). Fortunately, the particular form of the posterior distribution due to the absence of cycles in the dependency graph allows the application of optimization techniques similar to the ones used for the Viterbi algorithm [29][11]. Using dynamic programming, the best configuration can be computed in two recursive passes. The first bottom up pass calculates the maximum posterior probability of a labeling of site $s$ and its descendants $s_+$ given a labeling of the parent node $s^-$ (denoted as $\mu_s(x_{s^-})$) as follows:

$$\mu_s(x_{s^-}) = \max_{x_s} p(y_s|x_s)p(x_s|x_{s^-}) \prod_{t\in s_-} \mu_t(x_{s^-})$$

where the product over the children sites $t \in s_-$ is omitted for the leaf sites. The best label for each site, i.e. label resulting in the maximum probability, is stored:

$$\nu_s(x_{s^-}) = \arg\max_{x_s} p(y_s|x_s)p(x_s|x_{s^-}) \prod_{t\in s_-} \mu_t(x_{s^-})$$

The top down pass is initialized by the choice of the root label $x_r$:

$$\widehat{x_r} = \arg\max_{x_r} p(y_r|x_r)p(x_r) \prod_{s\in r_-} \mu_s(x_r)$$

The lower levels can be directly selected from the values stored in the bottom up pass:

$$\widehat{x_s} = \nu_s(x_{s^-})$$

## 4 The proposed cube model

The main disadvantage of the Markov quad tree is the non stationarity introduced into the random process of the leaf sites $G^{(0)}$ due to the fact that, at any given level, two neighboring sites may share a common parent or not depending on their position on the grid. In particular, for any two given neighboring leaf sites $s$ and $s'$, the minimum level $l$ containing a common ancestor of $s$ and $s'$ may be any value between 1 (the two nodes share an immediate parent) and $L-1$ (the root node $r$ is the only common ancestor of $s$ and $s'$) depending on their position on the grid. This lack of shift invariance causes visible "blocky" artifacts in the segmentation results.

We propose therefore a new model, which combines the advantages of causal hierarchical models with the shift invariance of stationary Markov random fields. The extension of the Markov quad tree to the Markov cube is shown in figures 2a-d, where for easier representation the one dimensional case is shown. In this case, the quad tree corresponds to a dyadic tree (figure 2a).

First, a second dyadic tree is added to the graph, which adds a common parent to all neighboring leaf sites which did not yet share a common parent in the original tree. In the full two dimensional case, three new quad trees are added. Note, that the graph now contains 2 root nodes in the 1D representation and 4 root nodes in the full 2D model. The problem is solved for the first level, where all neighboring sites share common parents. The number of parents increased from 1 to 2 (1D representation) or 4 (the full 2D model). The result of this step is seen in figure 2b. We repeat the process for each level, where at each level several new trees are added. The new trees connect sites of the original quad tree, but also sites of the trees added at the lower levels. The final result can be seen in figure 2d. Note, that the final graph is not a pyramid anymore, since each level contains the same number of nodes. In general, each node has 4 parents (2 in the 1D representation) a part from border nodes which may have less parents.

The whole graph can be efficiently implemented by a cube of dimensions $N \times N \times \mathrm{ld}(N)$, $N$ being the height/width of the image[2]. The parents and children of site $s$ having coordinates $x$ and $y$ on level $l$ are given as follows:

$$s^- = \begin{cases} x+\Delta^n, & y+\Delta^n, & l+1 \\ x+\Delta^n, & y+\Delta^p, & l+1 \\ x+\Delta^p, & y+\Delta^n, & l+1 \\ x+\Delta^p, & y+\Delta^p, & l+1 \end{cases}$$

$$s_- = \begin{cases} x+\Delta_n, & y+\Delta_n, & l-1 \\ x+\Delta_n, & y+\Delta_p, & l-1 \\ x+\Delta_p, & y+\Delta_n, & l-1 \\ x+\Delta_p, & y+\Delta_p, & l-1 \end{cases}$$

where

$$\Delta^n = \begin{cases} -1 & \text{if } l=0 \\ -2^{l-1} & \text{else} \end{cases} \qquad \Delta^p = \begin{cases} 0 & \text{if } l=0 \\ 2^{l-1} & \text{else} \end{cases}$$

$$\Delta_n = \begin{cases} 0 & \text{if } l=1 \\ -2^{l-2} & \text{else} \end{cases} \qquad \Delta_p = \begin{cases} 1 & \text{if } l=1 \\ 2^{l-2} & \text{else} \end{cases}$$

The probability distribution induced by the graph satisfies the following Markov condition:

$$p(x_s|x_{G\setminus s}) = p(x_s|x_{s^-}) \quad \forall s \in G$$

As for all causal hierarchical models, sampling from the joint probability distribution represented by the graph can be done in a single top down sweep, since the directed dependency graph does not contain cycles (taking into account the direction of the edges): labels for the top level nodes are sampled according to the prior distribution $p(x_s)$, and the labels for the nodes of the successive levels are sampled according to the conditional distributions $(x_s|x_s^-)$ and the labels of the respective parents.

The graph as it is described in figure 2d (in a 1D representation) corresponds to the hidden part, i.e. the prior model $p(x)$ in the Bayesian sense. In this work, as in the Markov quad tree model presented in section 3, we consider one observed node related to each hidden node and independence of the observed nodes conditional to the hidden nodes, i.e. the full joint probability distribution (including the observed nodes $Y_G$) induced by the graph satisfies equation (1) and factorizes as follows:

$$p(x,y) = \prod_{s\in G^{(0)}} p(x_s) \prod_{s\in G^{(l)},\ l>0} p(x_s|x_{s^-}) \prod_{s\in G} p(y_s|x_s) \tag{3}$$

As the Markov quad tree, the full Markov cube including observed nodes is parametrized through three probability distributions: the discrete prior distribution of the top level labels $p(x)$, the transition probabilities $p(x_s|x_{s^-})$ and the likelihood of the observed nodes given the corresponding hidden nodes $p(y_s|x_s)$ — a probability density.

For the inference algorithm, observations at different cube levels are needed. These observations may directly

[2]For ease of notation we assume images having equal height and width. The algorithm is, however, not subject to any restrictions.

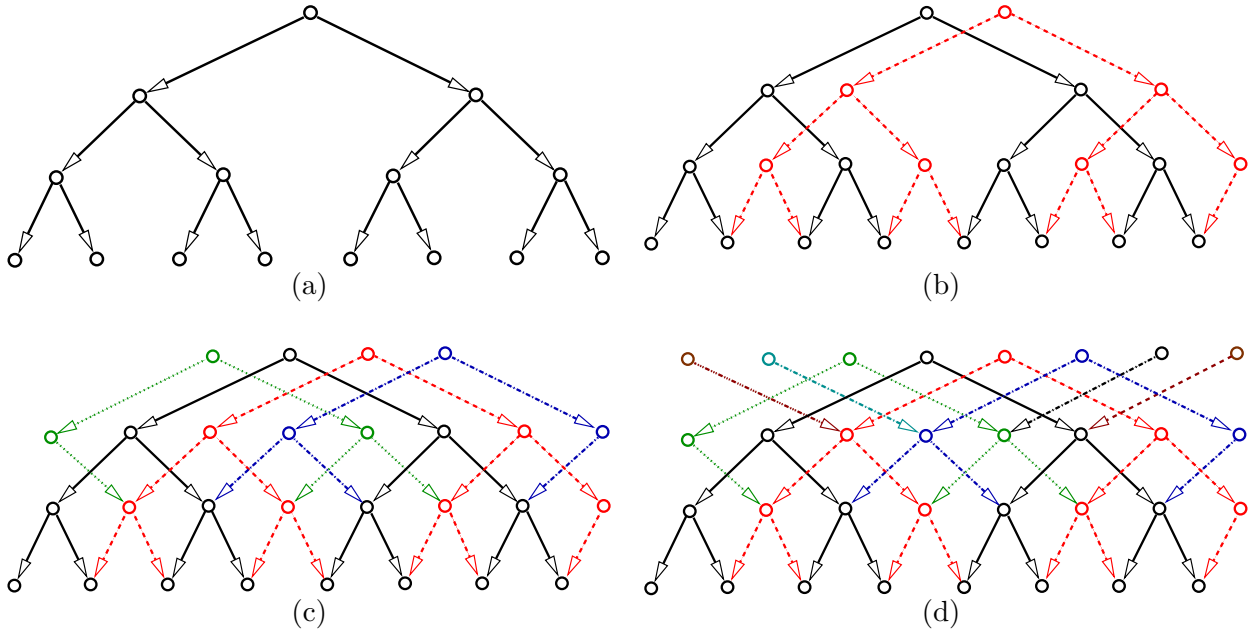


Figure 2: A one dimensional representation of the stepwise extension of the Markov quad tree [17] (represented as a dyadic tree) to a Markov cube (represented as a "Markov square"): a quad tree (represented as a dyadic tree) (a) after adding additional connections on the first level including the corresponding tree structure (b) after adding additional connections on the second level including the corresponding tree structure (c) the full Markov cube (d).

be taken from multi-resolution data, as long as the specific reduction function[3] of the graphical structure is taken into account. In most cases this will require resampling the data in all levels except the finest one. In image segmentation applications, the only available observations are at the base level ($y_{G^{(0)}}$). The higher levels can be calculated recursively, e.g. through a mean filter.

In image segmentation applications, the only available observations are at the base level ($y_{G^{(0)}}$). The higher levels can be calculated recursively through a mean filter:

$$d_s = \frac{1}{|s_-|} \sum_{s' \in s_-} d_{s'}$$

## 5 Inference with loopy belief propagation

The MAP estimator for the Markov cube is given as

$$\begin{aligned} \widehat{x} = & \\ = & \arg\max_{x \in \Omega} p(x)p(y|x) \\ = & \arg\max_{x \in \Omega} p(x,y) \\ = & \arg\max_{x \in \Omega} \prod_{s \in G^{(0)}} p(x_s) \prod_{s \in G^{(l)},\ l>0} p(x_s|x_{s^-}) \prod_{s \in G} p(y_s|x_s) \end{aligned} \tag{4}$$

Direct evaluation of equation (4) is intractable, and non iterative inference algorithms similar to the ones for the Markov quad tree are are made impossible by the cycles in the directed dependency graph (when the direction of the edges is not taken into account). Loopy belief propagation [22] is an approximative inference technique for general graphs with cycles. In practice, convergence does occur for many types of graph structures. Murphy et al. present an empirical study [21] which indicates that with LBP the marginals often converge to good approximations of the posterior marginals.

Loopy belief propagation is equivalent to the sum-product (or max-product) algorithm proposed for factor graphs [14]. Any directed or undirected dependency graph can be transformed into a factor graph which con-

[3]The term reduction function is taken from image pyramids where a set of nodes at one level is represented by ("reduced to") a single node at a higher level. In the case of the Markov cube where the number of nodes does not shrink at higher levels, a more adapted term could perhaps be "integration function".

tains two different types of nodes: variable nodes corresponding to the random variables of the model and factor nodes corresponding to the factors of the joint probability distribution. Figure 3 shows the 1D representation of a Markov cube without observed nodes as well as a small part of the full 2D Markov cube with observed nodes and their corresponding factor graphs.

The sum-product algorithm operates by passing messages between the nodes of the graph, each message being a function of the corresponding variable node. Due to the nature of our graph, there are two types of messages: messages from a variable node to a factor node, and the opposite:

- messages from a variable node $x_s$ upwards to the factor node.
- messages downwards to a variable node $x_s$ coming from the factor node.
- for each child $x_c$ of a variable node $x_s$, a message from $x_s$ downwards.
- for each child $x_c$ of a variable node $x_s$, a message upwards to $x_s$.

The message passing schedule for the cube alternates between bottom up passes and top down passes.

# 6 Interpretation of the hidden variables

In this section, we propose a methodology to estimate the conditional probability distributions $p(x_s|x_{s^-})$ by taking into account statistical invariance of images belonging to the same corpus. We propose to give an interpretation of the hidden variables $x_s$ (i.e. the variables belonging to level $l$>0) such that :

1. the independence model given by the structure is satisfied. Given a topological enumeration of vertices, a variable $x_s$ should be independent of all smaller index variables given its parents $x_s^-$.
2. the conditional probabilities are significantly different of conditional probabilities obtained on randomly binary images.
3. the conditional probabilities are close for all images of the corpus

For simplicity reasons, in the following we describe the binary case ($C = 2$), the adaptation to multiple labels is straightforward. Let $x_s$ be a vertex of the Markov cube and $l$ its level. We call $U_x$ the set of vertices of level 0 reachable by a directed path from $x$. $U_{x_s}$ is a $2^l * 2^l$ square on the image. Then, we naturally define the class of $x_s$ as the class with the maximum number of variables $U_{x_s}$(in case of equality, we choose the class randomly). In order to achieve estimation, we just have to compute the frequency of label 0 (resp 1) for each parent configuration. In our corpus, the 3 issues claimed above were verified. This interpretation allows several strategies for estimation of the conditional probabilities:

- Nonparametric definition of the conditional probabilities. Given initial labels at the base level, the labels at the upper levels are computed as described above and the probabilities are estimated using histograming.
- Parametric functions are fitted to the histograms. This strategy is pursued in the next section.

# 7 Inference with graph cuts

Algorithms calculating the minimum cut/maximum flow in a graph are a powerful tool able to calculate the *exact* MAP solution on a number of binary labeling problems [8, 4, 5, 13] with low order polynomial complexity. It has been shown recently, that energy functions for which the optimal solution is equivalent to the minimum cut in an appropriate graph contain only "regular" terms on binary labels [13], where regular means that any projection of a term $E(x_i, x_j, x_k, \ldots)$ onto any subset of two of its arguments satisfies the following condition:

$$E(0,0) + E(1,1) \leq E(0,1) + E(0,1) \tag{5}$$

In the case of the proposed model, not all energy terms are regular, especially the terms corresponding to the logarithm of the transition probabilities $\ln p(x_s|x_{s^-})$, so the general model cannot be solved with graph cuts. However, for a large sub class with interesting properties, graph cut solutions can be found. We propose a regularizing term based on the number of parent labels which are equal to the child label:

$$p(x_s|x_{s^-}) = \frac{1}{Z}\alpha_l^{\xi(x_s,x_{s^-})} \tag{6}$$

where $\alpha_l$ is a parameter depending on the level $l$, $\xi(x_s, x_{s^-})$ is the number of labels in $x^-$ equal to $x_s$ and $Z$ is a normalization constant. The such defined transition probabilities favor homogeneous regions, which corresponds to the objective of an image segmentation

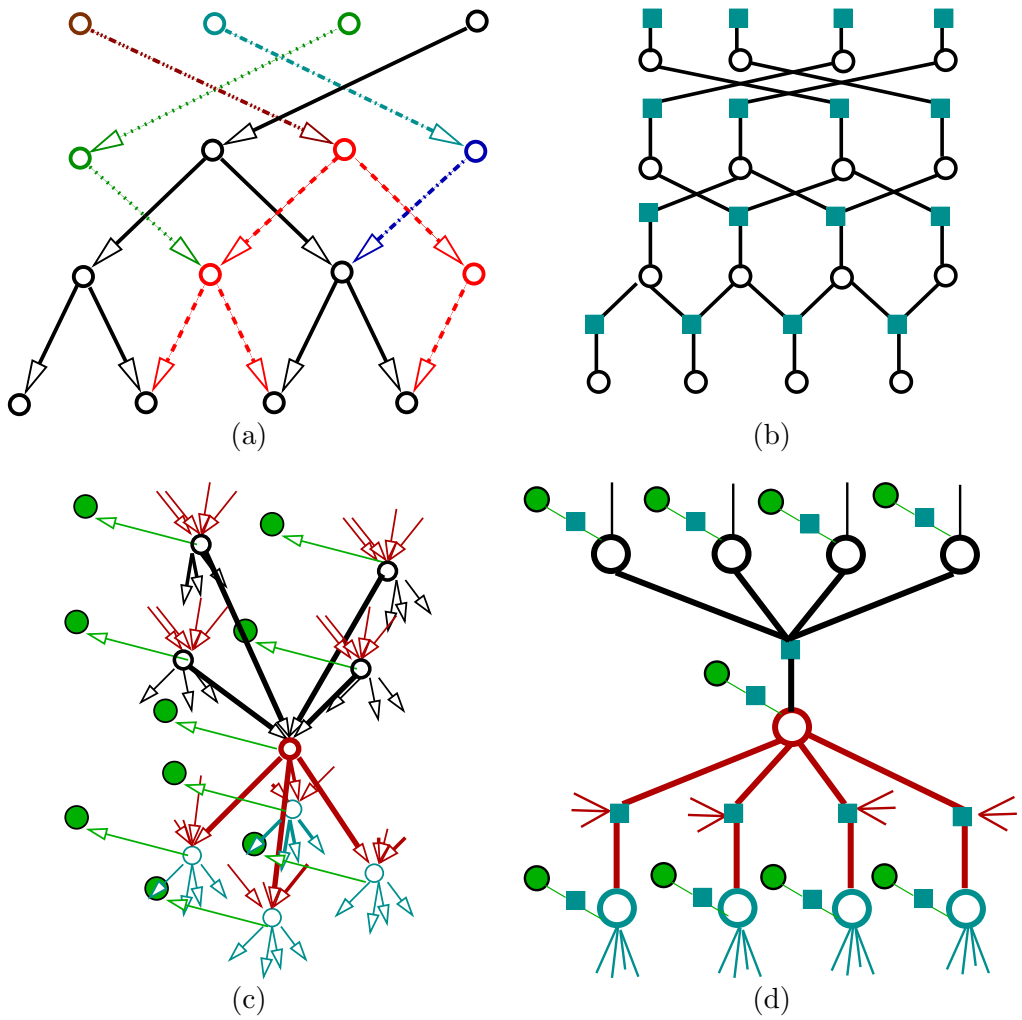


Figure 3: A cube (left) and its factor graph (right) in one dimension (top) and two dimensions (bottom)

algorithm. We then decompose it into a sum of binary terms:

$$\ln p(x_s|x_{s^-}) = \sum_{s' \in s^-} [(\ln \alpha)\ \delta_{x_s,x_{s'}}] - Z \qquad (7)$$

where $\delta_{a,b}$ is the Kronecker delta defined as 1 if $a = b$ and 0 else. It should be noted that each binary term is regular in the sense of [13]. Fig. 4 shows a cut graph constructed for the dependency graph of Fig. 3b: the cut with minimum cost separating source $S$ from sink $T$ corresponds to the exact MAP estimate for a Markovcube with binary labels ($C = 2$). Each non terminal node is connected to one of the terminal nodes with weight $|\ln p(y_s|x_s = 1)/p(y_s|x_s = 0)|$, according to the sign inside the absolute value. The weights of top level nodes $s$ contain an additional term $\ln p(x_s = 1)/p(x_s = 0)$. Additionally, each non terminal node is connected to each of its parents with an undirected edge and weight $\ln \alpha$.

Minimum cut algorithms are restricted to binary labeling problems ($C = 2$). Discontinuity preserving energy minimization with multiple labels is NP-hard [5], but the $\alpha$-expansion move algorithm introduced in [5] allows to find a local minimum with guaranteed maximum distance to the global minimum. It consists of iteratively applying the minimum cut algorithm to the sub problem of labeling each node of the whole graph between two labels: keeping the current label and changing the a new label $\alpha$, which is changed at each iteration.

## 8 Parameter estimation

We chose the unsupervised technique Iterated Conditional Estimation (ICE) [23] for parameter identification. Given supervised estimators of the parameters from a realization of the full set of variables $(X, Y)$, an iterative procedure alternates between estimating the parameters and creating realizations of the label field based on the current parameters. The initial set of parameters can be obtained from an initial segmentation of the input image.

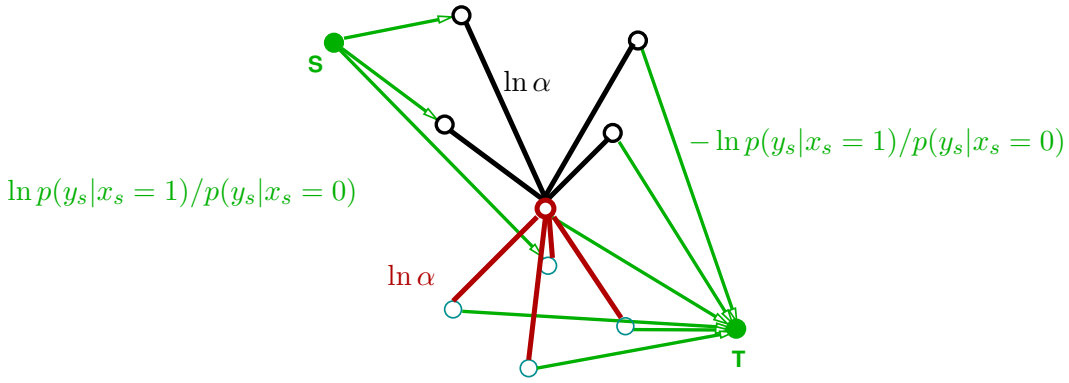


Figure 4: The cut graph constructed for the binary problem from the dependency graph shown in Fig. 3b, including the two terminal nodes $S$ and $T$. For more than 2 labels, the expansion move algorithm ressorts to a similar graph.

The prior probabilities of the top level labels $\beta_i$ can be estimated using histogram techniques. Similarly, for most common observation models, maximum likelihood estimators of the sufficient statistics of the conditional distributions are readily available. In this paper, we work with a simple observation model assuming Gaussian noise, requiring as parameters means and (co)-variances for each class. Arbitrary complex likelihood functions are possible using Gaussian mixtures.

For the parameters $\alpha_l$ of the transition probabilities, we propose a solution based on least squares estimation similar to the works proposed by Derin et al. for the estimation of Markov random field parameters [7]. For each level $l$, we consider pairs of different site labels $x_s$ and $x_{s'}$ ($s \in G^{(l)}$) with equal parent labels $x_{s^-} = x_{s'^-}$. Note that the parent sites are different, whereas their labels are equal. From (6) the following relationship can be derived:

$$\frac{P(x_s|x_{s^-})}{P(x_{s'}|x_{s^-})} = \frac{\alpha_l^{\xi(x_s,x_{s^-})}}{\alpha_l^{\xi(x_{s'},x_{s^-})}} \tag{8}$$

Expressing the conditional probabilities through absolute probabilities and taking the logarithm we get:

$$\ln \alpha_l [\, \xi(x_s, x_{s^-}) - \xi(x_{s'}, x_{s^-}) \,] = \ln \left[ \frac{P(x_s, x_{s^-})}{P(x_{s'}, x_{s^-})} \right] \tag{9}$$

The right hand side of the equation can be estimated from the label process, e.g. by histogramming, whereas the factor in the left hand side can be calculated directly. Considering a set of different label pairs, we can augment this to

$$\boldsymbol{b}^T [\ln \alpha_l] = \boldsymbol{a} \tag{10}$$

where $\boldsymbol{b}$ is a vector where each element corresponds to the value in the left hand side of equation (9) for a given label pair and each value in the vector $\boldsymbol{a}$ corresponds to the right hand side of equation (9) for a given label pair. The solution of the over determined linear system can be found using standard least squares techniques.

## 9 Complexity and storage

Inference complexity for loopy belief propagation (LBP) can be given as $O(I \cdot N \cdot M \cdot (H-1) \cdot C^5)$ where $I$ is the number of iterations. $H$ is the height of the cube and bounded by $\lceil \log_2 \max(N, M) \rceil$. Storage requires $N \cdot M \cdot (H-1) \cdot 15C$ variables. In practice, LBP in its original form is applicable for low numbers of classes (2, 3 or maximum 4), which is enough for a large number of problems. For higher numbers of classes, the classes may be quantized and the message passing equations slightly changed.

Inference with minimum cut/maximum flow is considerably faster with a complexity bounded by $O(E * f)$, where $E$ is the number of edges in the graph and $F$ is the maximum flow. We use the graph cut implementation by Boykov and Kolmogorov [4] which has been optimized for typical graph structures encountered in computer vision and whose running time is nearly linear in running time in practice [5]. Table 1 gives effective run times and memory requirements measured on a computer equipped with a single core Pentium-M processor running at 1.86Ghz.

## 10 Experimental results

We evaluated the model on synthetic data as well as real scanned images. In all experiments, we initialized the label field with k-means clustering after low pass filtering.

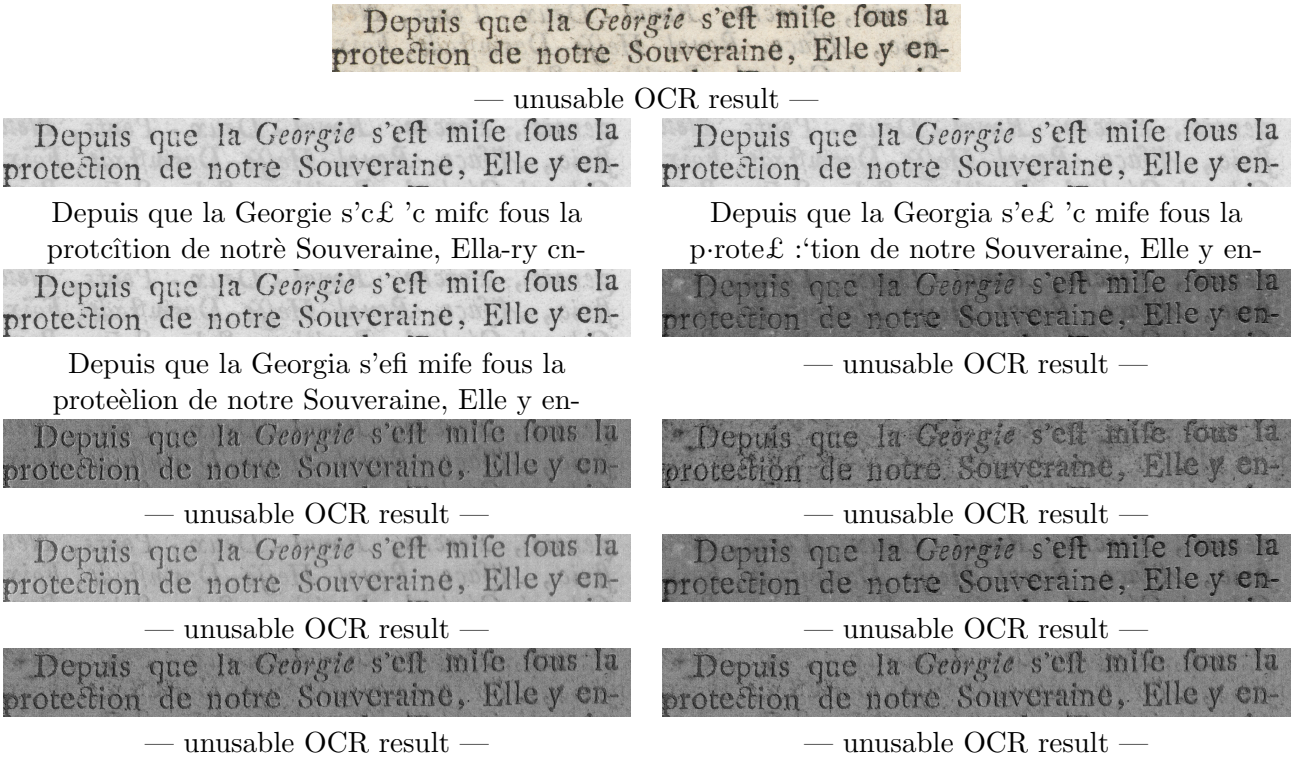


Figure 6: Restoration and OCR results on real data, from left to right, top to bottom: input image, k-means, MRF[13], markovcube, 4× Tonazzini et. al [27] (plane #1, plane #2, plane #3, all 3 planes combined), 2× Tonazzini et al. [28] (plane #1, plane #2).

| Method | MB | sec. |
|---|---|---|
| K-means | 1 | 1 |
| Quad tree | 5 | 1 |
| MRF-GC | ∼20 | 2 |
| Cube-LBP (4 levels, non-param.) | 103 | 46 |
| Cube-LBP (4 levels, parametric) | 103 | 46 |
| Cube-LBP (5 levels, parametric) | 150 | 64 |
| Cube-GC (5 levels, parametric) | ∼180 | 4 |

Table 1: Execution times and memory requirements of different algorithms.

| Method | Error rate |
|---|---|
| K-means | 27.01 |
| K-means (incl. low pass filter) | 9.01 |
| Quad tree | 7.57 |
| MRF-GC | 6.28 |
| Cube-LBP (4 levels, non-parametric) | 6.82 |
| Cube-LBP (4 levels, parametric) | 6.91 |
| Cube-LBP (5 levels, parametric) | 6.84 |
| Cube-GC (5 levels, parametric) | **5.58** |

Table 2: Pixel level segmentation performance on synthetic images of size 512×512 and ($C$=2)

### 10.1 Pixel level evaluation

To be able to evaluate the model's segmentation performance quantitatively, we applied it to 30 synthetic images of size 512×512 (60 images total) and very low quality subject to multiple degradations: low pass filtering, amplification of ring shaped frequency bands causing ringing artifacts, low quality JPEG artifacts and additional Gaussian noice in various stages (with variances between $\sigma$=20 and $\sigma$=40). We compared the cube model with different methods of the state of the art: flat MRF segmentation with a Potts model and graph cut optimization [13], a quad tree [17] and k-means clustering. The k-means algorithm is only method whose performance is improved when the image is low pass filtered before the segmentation. Table 2 shows the error rates on the different sets.

### 10.2 Measuring OCR improvement

To further evaluate our algorithm we tested it on a real application, namely the restoration of images degraded with ink bleedthrough. The goal is to remove the verso component from the recto scan, which makes it a three class segmentation problem. We chose a dataset consisting of 6 images of pages scanned with 600dpi con-

Figure 5: Zoom into the results shown in figure 6: (top) segmentation result (bottom) restoration result (left) MRF (right) markovcube.

| Method | | Recall | Precision |
|---|---|---|---|
| No restoration | † | - | - |
| K-Means (k=3) | | 61.23 | 51.74 |
| Tonazzini et al. [28] | † | - | - |
| Tonazzini et al. [27] | † | - | - |
| Tonazzini et al. [27] | § | 13.13 | 25.43 |
| MRF [13] | | 69.10 | 58.42 |
| Simple markovcube | | **69.34** | **61.19** |

†Not available: lack of OCR performance makes a correct evaluation impossible

§Results obtained combining all source planes

Table 3: Evaluation of the character recognition (OCR) improvement caused by different restoration methods when applied to scanned document images.

taining low quality printed text from the $18^{th}$ century, the *Gazettes de Leyde*, a journal in French language printed from 1679 to 1798. We tested our method's ability to improve the performance of an OCR algorithm and compared it to several widely cited algorithms: k-means clustering, a flat Markov random field (MRF) with graph cuts optimization [13], as well as two well known methods[4] based on source separation [27, 28].

Figure 6 shows parts of the images together with the OCR results. We manually created groundtruth and calculated the recall and precision measures on character level, which are given in table 3.

As we can see, our general purpose model outperformes all other segmentation algorithms. The flat MRF directly models the interactions between pixels, which in theory is more powerful than the scale interactions of the markov cube. However, this is only interesting in cases where no long run interactions are needed, e.g. in images with small structures. In images with larger and, more importantly, scale varying content, the hierarchical nature of the markov cube manages to better model the image contents, which directly translates into a better restoration segmentation and restoration performance.

Surprisingly, the recognition performance on the results of the two source separation results was very disappointing. Unfortunately, the recognition performance on these results was not good enough to include it in the table. Most of the output was blank or gibberish, making an evaluation impossible.

Figure 5 shows a zoom into the results comparing the flat MRF and the markov cube. As we can see, the hierarchical nature of the cube results in a better segmentation performance by removing artifacts and filling holes.

## 11 Conclusion and discussion

In this paper we presented a new causal model which features the advantages of hierarchical models, i.e. scale dependent behavior and the resulting adaptivity to the image characteristics, without the main disadvantage of the quad tree model, i.e. the lack of shift invariance. Bayesian maximum a posteriori estimation on this model has been tested on binarization and ink bleed through removal tasks for document images and compared to widely used graphical models. Segmentation quality is better or equal to the results of a MRF model, the difference depending on the scale characteristics of the input image and the nature of the degradation. We proposed two inference algorithms: loopy belief propagation and an algorithm based on graph cuts for regular transition probability distributions.

## References


[1] K. Abend, T.J. Harley, and L.N.Kanal. Classification of binary random patterns. *IEEE Transactions on Information Theory*, IT-11(4):538–544, 1965.

[2] M.G. Bello. A combined Markov random field and wave-packet transform-based approach for image segmentation. *IEEE transactions on image processing*, 3(6):834–846, 1994.

[3] C.A. Bouman and M. Shapiro. A Multiscale Random Field Model for Bayesian Image Segmentation. *IEEE Transactions on Image Processing*, 3(2):162–177, 3 1994.


[4]We thank Anna Tonazzini for providing us with the source code of the two source separation methods and her kind help in setting up the corresponding experiments as well as for the interesting discussions.

[4] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[5] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[6] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[7] H. Derin and H. Elliott. Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1):39–55, 1987.

[8] D.M.Greig, B.T. Porteous, and A.H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society B*, 51(2):271–279, 1989.

[9] R. Fjortoft, Y. Delignon, W. Pieczynski, M. Sigelle, and F. Tupin. Unsupervised classification of radar images using hidden Markov chains and hidden Markov random fields. *IEEE Transaction on Geoscience and remote sensing*, 41(3):675–686, 2003.

[10] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 11 1984.

[11] G.D. Fornay Jr. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[12] Z. Kato, M. Berthod, and J. Zerubia. A hierarchical Markov random field model and multitemperature annealing for parallel image classification. *Graphical Models and Image Processing*, 58(1):18–37, 1996.

[13] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[14] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE transactions on Information Theory*, 47(2):498–519, 2001.

[15] S. Kumar and M. Hebert. Discriminative random fields. *International Journal of Computer Vision*, 68(2):179–201, 2006.

[16] S.-S. Kuo and O.E. Agazzi. Keyword spotting in poorly printed documents using pseudo 2-d hidden Markov models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):842–848, 1994.

[17] J.-M. Laferte, P. Perez, and F. Heitz. Discrete Markov image modelling and inference on the quad tree. *IEEE Transactions on Image Processing*, 9(3):390–404, 2000.

[18] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling data. In *International Conference on Machine Learning*, 2001.

[19] E. Levin and R. Pieraccini. Dynamic planar warping for optical character recognition. In *Proceedings of the International Conference on Accoustics, Speech and Signal Processing*, volume 3, pages 149–152, 1992.

[20] M. Mignotte, C. Collet, P. Perez, and P. Bouthemy. Sonar image segmentation using an unsupervised hierarchical mrf model. *IEEE Transactions on Image Processing*, 9(7):1216–1231, 2000.

[21] K. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief-propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 467–475, 1999.

[22] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufman, San Mateo, 1988.

[23] W. Pieczynski. Convergence of the iterative conditional estimation and application to mixture proportion identification. In *IEEE/SP Workshop on Statistical Signal Processing*, pages 49–53, 2007.

[24] W. Pieczynski and A.-N. Tebbache. Pairwise Markov random fields and segmentation of textured images. *Machine Graphics & Vision*, 9(3):705–718, 2000.

[25] A. Rosenfeld. The prism machine: an alternative to the pyramid. *Journal of Parallel and Distributed Computing*, 2(4):404–411, 1985.

[26] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2006.

[27] A. Tonazzini and L. Bedini. Independent component analysis for document restoration. *International Journal on Document Analysis and Recognition*, 7(1):17–27, 2004.

[28] A. Tonazzini, E. Salerno, and L. Bedini. Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique. *International Journal on Document Analysis and Recognition*, 10(1):17–25, 2007.

[29] A.J. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory*, IT-13:260–269, 1967.

[30] Y. Wang, K.-F. Loe, and J.-K. Wu. A dynamic conditional random field model for foreground and shadow segmentation. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 28(2):279–289, 2006.

[31] R. Wilson and C.-T. Li. A class of discrete multiresolution random fields and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):42–56, 2002.