Inference and parameter estimation on hierarchical belief networks for image segmentation

Christian Wolf, Gérald Gavin

Inference and parameter estimation on hierarchical belief networks for image segmentation

Christian Wolf[1,2] and Gérald Gavin[1,3]

[1]University of Lyon, CNRS, UMR5205
[2]INSA-Lyon, LIRIS, UMR5205, F-69621, France
[3]Université Lyon 2, ERIC, F-69676, France
christian.wolf@liris.cnrs.fr, Gerald.Gavin@univ-lyon1.fr

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 documents, showing an improvement of character recognition results compared to other methods.

Keywords

Belief networks, image segmentation, graph cuts

1 Introduction

Image segmentation techniques aim at partitioning images into a set of non overlapping and homogeneous regions taking into account prior knowledge on the results as well as a probabilistic model of the observation (degradation) process. Belief networks, but also undirected probabilistic graphical models, are widely used to incorporate spatial dependencies between the pixels into the classification process, often formulating the problem as Bayesian estimation.

In their seminal paper [8], Geman and Geman introduced a maximum a posteriori (MAP) estimation technique for Markov random fields (MRF). An alternative to the two-dimensional MRFs are hidden Markov chains (MC) on one-dimensional traversals (Hilbert-Peano scans) of an image [1] or hybrid MC/MRF techniques [7]. The Markov chain models have been extended to belief networks with a pseudo 2D structure [12] and to full 2D connectivity [14].

Hierarchical models introduce a scale dependent component into the classification algorithm, which allows the algorithm to better adapt itself to the image characteristics. The nodes of the graph are partitioned into different scales, where lower scale levels correspond to finer versions of the image and higher scale levels correspond to coarser versions of the image. Lower scales manage interactions on pixel level, whereas higher scales manage interactions of groups of pixels (regions). Examples are stacks of flat MRFs [2], pyramidal graph structures [9] and the scale causal multi-grid [15]. Bouman and Shapiro were among the first to propose a hierarchical belief network for image segmentation [3] (refined by Laferte *et al.* [13]). A quad tree models the spatial interactions between the leaf pixel sites through their interactions with neighbors in scale. The main problem of the quad tree structure is the non stationarity it induces into the random process of the leaf sites, which results in "blocky" artifacts in the segmented image.

In the same paper a second model is proposed, 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 pyra-

midal model in that the number of nodes decreases at each level. Moreover, as an approximation the inference algorithm proposes a change of the graph after each inference step, whereas in our work the whole graph keeps its full connectivity.

The work described in this paper concentrates on the solution to the lack of shift invariance of the quad tree structured network. 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 2 describes the quad tree structured network and section 3 extends it to the cube. Section 4 presents an inference algorithm using loopy belief propagation and section 5 outlines an interpretation of the hidden variables of the model. Section 6 presents a fast inference algorithm for a parametric class of transition probability distributions. Section 7 describes parameter estimation for the latter class of distributions and section 8 deals with the computational complexity and memory requirements. Section 9 experimentally validates the method. Finally, section 10 concludes.

2 Quad tree structured models

In the following we describe belief networks, thus graphical models defined on directed acyclic graphs $\mathcal{G} = \{G, E\}$, 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 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. 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 denoted as $\Omega = \Lambda^{|G|}$.

In the case of the quad tree structured model [3, 13], 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 Fig. 1a). Each hidden variable X_s is related to an observed variable Y_s which is conditionally independent of the other observed variables given a realization of X_s: $P(y_s|x) = P(y_s|x_s)$ and $P(y|x) = \prod_{s \in G} P(y_s|x_s)$ (see Fig. 1b).

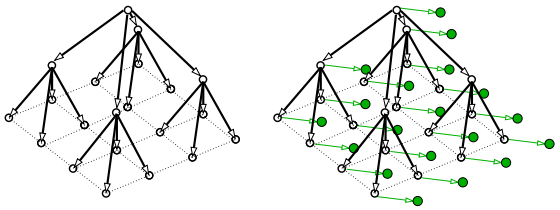

Figure 1: The quad tree structured model with and without observed nodes.

The objective is to estimate the hidden variables x given the observed variables y, in the case of the MAP estimator this is done maximizing the posterior distribution: $\widehat{x} = \arg\max_{x \in \Omega} p(x|y)$. The absence of cycles in the dependency graph allows the application of an extension of the Viterbi algorithm [21, 13].

3 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. We therefore propose an extension shown in figures 2a-d (for easier representation the one dimensional case — a dyadic tree — is shown)

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 full 2D case, three new quad trees are added. The problem is solved for the first level, where the number of parents increased to 4 (for the full 2D model). The result of this step is seen in figure 2b. We repeat the process for each level. 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) except border nodes.

The whole graph can be efficiently implemented by a cube of dimensions $N \times M \times \lceil \log_2 \max(N, M) \rceil$, $N \times M$ being the size of the image. In practice, the full height of the cube is not always required, althought a certain number of levels is needed to ensure a proper amount of high-level interaction. The parents and

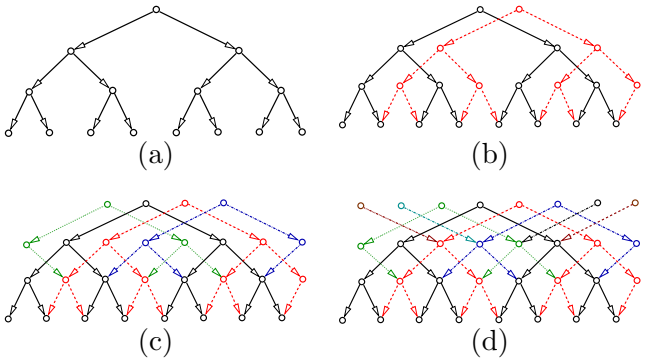

Figure 2: A one dimensional representation of the stepwise extension of the quad tree [13] (shown as a dyadic tree) to the proposed cube model

children of site s with coordinates x and y on level l are given as follows:

$$s^- = \begin{cases} x+\Delta^n, y+\Delta^n \\ x+\Delta^n, y+\Delta^p \\ x+\Delta^p, y+\Delta^n \\ x+\Delta^p, y+\Delta^p \end{cases} \quad s_- = \begin{cases} x+\Delta_n, y+\Delta_n \\ x+\Delta_n, y+\Delta_p \\ x+\Delta_p, y+\Delta_n \\ x+\Delta_p, y+\Delta_p \end{cases}$$

where

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

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

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). 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. As for the quad tree, 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 factorizes as follows:

$$p(x,y) = \prod_{s\in G^{(0)}} p(x_s) \prod_{s\in G^{(l)},\ l>0} p(x_s|x_{s^-}) \prod_{s\in G} p(y_s|x_s) \tag{1}$$

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 $p(y_s|x_s)$.

For the inference algorithm, observations at different cube levels are needed. These observations may directly be taken from multi-resolution data, as long as the specific reduction function 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.

4 Inference with loopy belief propagation

The MAP estimator maximizes equation (1), which is hard if done directly. Indeed, the graph of the model contains cycles when the direction of the edges is not taken into account. In other words, the factor graph of the model contains cycles, which makes an exact calculation using message passing algorithms intractable. The junction tree algorithm is capable of calculating the exact solution on general graphs, however, its complexity is exponential in the maximal clique size of the graph after moralization and triangulation. Calculating the optimal triangulation of a general graph being NP-complete. We applied heuristics to find good triangulations of the graph, which did not manage to provide us with tractable clique sizes, i.e. clique sizes which are independent of the size of the graph. Looking at the energy function derived from the (1), i.e. its negative logarithm, Kolmogorov and Zabih showed [10] that the minimization of the general form of energy functions involving terms of second order or higher is NP-complete, unless the energy potentials are submodular and the number of labels is equal to two[1] .

Loopy belief propagation (LBP) is an approximative inference technique for general graphs with cycles [17]. In practice, convergence does occur for many types of graph structures. Murphy et al. present an empirical study [16] 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

[1] Note that the mode of posterior marginals (MPM) estimator [13] is equally difficult to calculate due to the presence of cycles in the dependency graph. Although the partial posterior marginals $p(x_s|y_d(s))$, where $d(s)$ denotes the set of all descendants of s, are computable in $O(C^8)$, i.e. with high complexity but still in polynomial time, the computation of the posterior marginals $p(x_s|y)$ is difficult due to the lack of a common root node which could start a recursion.

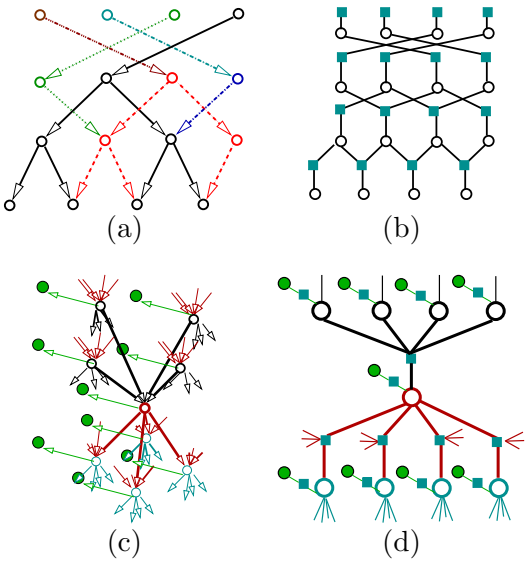

Figure 3: A cube (left) and its factor graph (right) in one dimension (top) and two dimensions (bottom)

factor graphs [11]. Any directed or undirected dependency graph can be transformed into a factor graph, and 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. The message passing schedule for the cube alternates between bottom up and top down passes.

5 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.

6 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 [4, 5, 10] 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 submodular terms on binary labels [10], where submodular means that any projection of a term $E(x_i, x_j, x_k, \dots)$ onto any subset of two of its arguments satisfies the following condition[2]:

$$E(0,0) + E(1,1) \leq E(0,1) + E(1,0) \qquad (2)$$

In the case of the proposed model, not all energy terms are submodular, 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:

$$p(x_s|x_{s^-}) = \frac{1}{Z}\alpha_l^{\xi(x_s,x_{s^-})} \qquad (3)$$

[2]In [10], the term *regular* is used instead of *submodular*

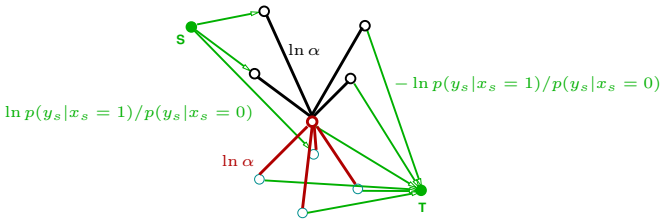

Figure 4: The cut graph constructed for the binary problem from the dependency graph shown in Fig. 3b, including the two terminal nodes S and T.

where α_l is a parameter depending on the level l, $\xi(x_s, x_{s^-})$ is the number of labels in x_{s^-} 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 algorithm. We then decompose it into a sum of binary terms:

$$\ln p(x_s|x_{s^-}) = \sum_{s' \in s^-} [(\ln \alpha)\ \delta_{x_s, x_{s'}}] - Z \qquad (4)$$

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 submodular since the $\delta_{a,b}$ is submodular for all a, b. 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 α-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 α, which is changed at each iteration.

7 Parameter estimation

We chose the unsupervised technique Iterative Conditional Estimation (ICE) [18] for parameter identification. Given supervised estimators of the parameters from a realization of the full set of variables (X, Y), an iterative procedure estimates a new set of parameters as the conditional expectation of the parameters conditioned on the observed field Y and the current parameters. In pratice, the expectations are hard to calculate but can be approximated by simulations 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.

The prior probabilities of the top level labels β_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 α_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 [6]. 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 (3) the following relationship can be derived:

$$\frac{P(x_s|x_{s^-})}{P(x_{s'}|x_{s^-})} = \frac{\alpha_l^{\xi(x_s, x_{s^-})}}{\alpha_l^{\xi(x_{s'}, x_{s^-})}} \qquad (5)$$

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

$$\ln \alpha_l [\ \xi(x_s, x_{s^-}) - \xi(x_{s'}, x_{s^-})\ ] = \ln \left[\frac{P(x_s, x_{s^-})}{P(x_{s'}, x_{s^-})} \right] \qquad (6)$$

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 $\boldsymbol{b}^T [\ln \alpha_l] = \boldsymbol{a}$ where $\boldsymbol{b}$ is a vector where each element corresponds to the value in the left hand side of equation (6) for a given label pair and each value in the vector $\boldsymbol{a}$ corresponds to the right hand side of equation (6) for a given label pair. The solution of the over determined linear system can be found using standard least squares techniques.

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)

Figure 5: Zoom into the results shown in figure 6: segmentation and restoration results for MRF (left) and markovcube+GC (right).

Method		Recall§	Precision§
No restoration	†	-	-
K-Means (k=3)		61.23	51.74
Tonazzini et al. [20]	†	-	-
Tonazzini et al. [19]	†	-	-
Tonazzini et al. [19]	‡	13.13	25.43
MRF [10]		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

§Recall=number of correctly recognized characters divided by number of correct characters in the groundtruth; Precision=number of correctly recognized characters divided by number of recognized characters.

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

8 Complexity and storage

Inference complexity for LBP can be given as $O(I \cdot N \cdot M \cdot (H-1) \cdot C^5)$ where I is the number of iterations and 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.

9 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.

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 σ=20 and σ=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 [10], a quad tree [13] 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, and figure 7 shows a zoom on an example image.

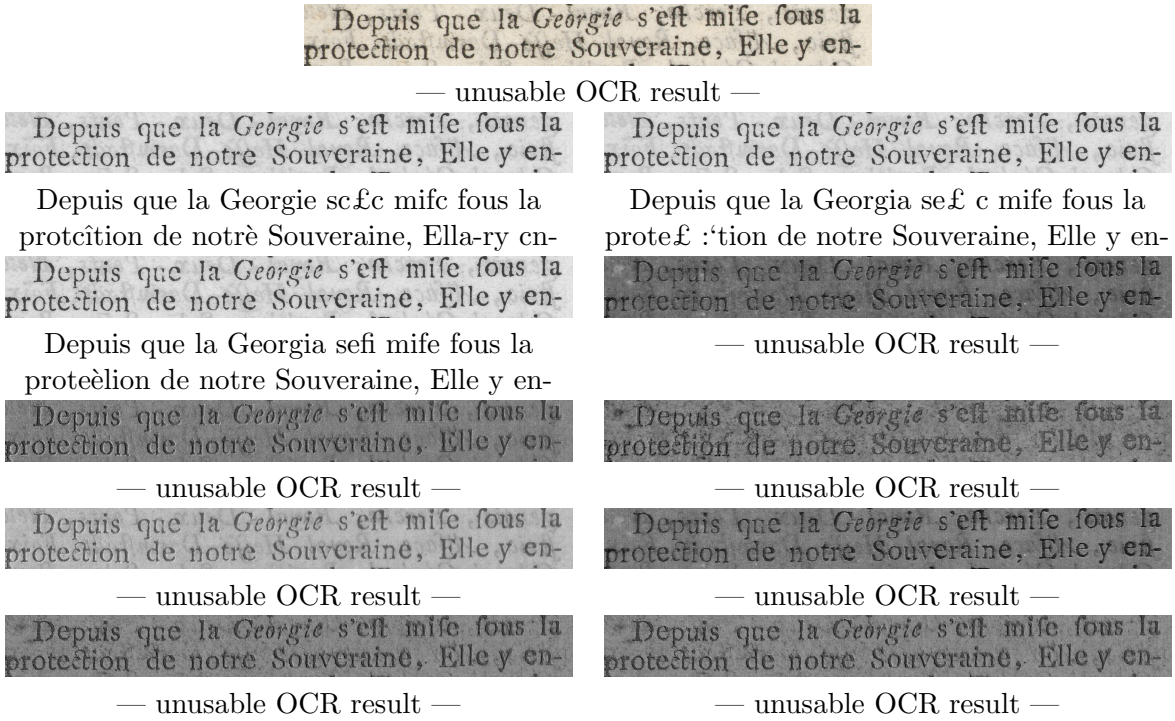

Figure 6: Restoration and OCR results on real data, from left to right, top to bottom: input image, k-means, MRF[10], markovcube+GC, 4× Tonazzini et. al [19] (plane #1, plane #2, plane #3, all 3 planes combined), 2× Tonazzini et al. [20] (plane #1, plane #2).

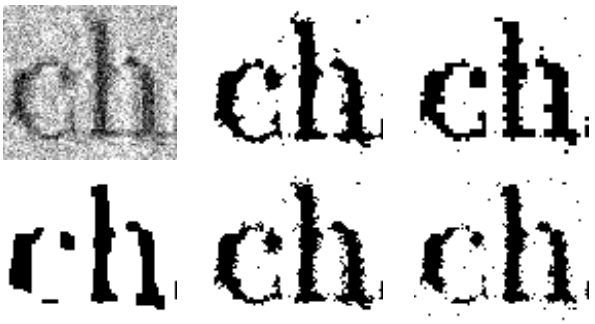

Figure 7: Zoom on binarization (C=2) results on synthetic images. From top to bottom, left to right: input image, kmeans+filtering, quad tree, MRF-GC, Cube-LBP, Cube-GC.

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 containing 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 optical character recognition (OCR) algorithm and compared it to several widely cited algorithms: k-means clustering, a flat Markov random field (MRF) with graph cuts optimization [10], as well as two well known methods[3] based on source separation [19, 20].

Figure 6 shows parts of the images together with the OCR results. As can be seen, the cube tends to regularize stronger, which smoothes the region boundaries. 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 outperforms 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

[3]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.

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 two source separation results performed so poorly that we were unable include their recognition performance in the table, since most of the output was blank or gibberish.

Figure 5 shows a zoom into the results comparing the flat MRF and the markov cube and shows that the hierarchical nature of the cube results outperforms the MRF, removes artifacts and fills holes.

10 Conclusion and discussion

We presented a causal model featuring 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 MAP estimation on this model has been tested on binarization and ink bleed through removal tasks and compared to widely used methods. Segmentation quality is better or equal to the results of a MRF model, the difference depending on the scale characteristics of the input image. LBP and a graph cut based algorithm are proposed for inference. Finally, let's note that the model can be extended to 3D data trivially, albeit with an increase in complexity.

References

[1] K. Abend, T.J. Harley, and L.N.Kanal. Classification of binary random patterns. *IEEE Tr. 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 tr. on image proc.*, 3(6):834–846, 1994.

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

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

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

[6] H. Derin and H. Elliott. Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. *IEEE Tr. on PAMI*, 9(1):39–55, 1987.

[7] 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.

[8] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. 6(6):721–741, 11 1984.

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

[10] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Tr. on PAMI*, 26(2):147–159, 2004.

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

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

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

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

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

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

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

[18] 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.

[19] A. Tonazzini and L. Bedini. Independent component analysis for document restoration. *I.J. on Doc. Anal. and Rec.*, 7(1):17–27, 2004.

[20] A. Tonazzini, E. Salerno, and L. Bedini. Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique. *I.J. on Doc. Anal. and Rec.*, 10(1):17–25, 2007.

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