Statistical Priors for Efficient Combinatorial Optimization Via Graph Cuts

Daniel Cremers[1] and Leo Grady[2]

[1] Department of Computer Science, University of Bonn, Germany
[2] Department of Imaging and Visualization,
Siemens Corporate Research, Princeton, NJ

Abstract. Bayesian inference provides a powerful framework to optimally integrate statistically learned prior knowledge into numerous computer vision algorithms. While the Bayesian approach has been successfully applied in the Markov random field literature, the resulting combinatorial optimization problems have been commonly treated with rather inefficient and inexact general purpose optimization methods such as Simulated Annealing. An efficient method to compute the global optima of certain classes of cost functions defined on binary-valued variables is given by graph min-cuts. In this paper, we propose to reconsider the problem of statistical learning for Bayesian inference in the context of efficient optimization schemes. Specifically, we address the question: Which prior information may be learned while retaining the ability to apply Graph Cut optimization? We provide a framework to learn and impose prior knowledge on the distribution of pairs and triplets of labels. As an illustration, we demonstrate that one can optimally restore binary textures from very noisy images with runtimes on the order of a second while imposing hundreds of statistically learned constraints per pixel.

1 Introduction

In his 1948 paper, Shannon considered the formation of text as a stochastic process. He suggested to learn the probabilities governing this process by computing the histograms of occurrences and co-occurrences of letters from a sample text. Subsequently he validated the accuracy of the generated model by sampling new texts from the estimated stochastic model. Not surprisingly, the successive integration of higher order terms (occurrence of letter triplets rather than pairs etc.) provides for the emergence of increasingly familiar or meaningful structures in the synthesized text.

In the context of images, similar approaches have been proposed in the Markov random field literature. We refer to [24] for an excellent introduction. Going back at least as far as Abend's work [1], Markov random fields have endured a sustained interest in the vision community. Besag [3] applied them in the context of binary image restoration and Derin [8] and Gimelfarb and coworkers [12] analyzed texture in the context of a Markov random field using learned priors based on gray level co-occurrences. Work has continued through new applications such as texture segmentation [20] or through extension of the basic model, for example by considering higher-order cliques [23].

There are two major computational challenges arising in the application of Markov random fields for Bayesian inference. Firstly, one needs to devise methods to efficiently

A. Leonardis, H. Bischof, and A. Pinz (Eds.): ECCV 2006, Part III, LNCS 3953, pp. 263–274, 2006.

learn priors given a set of representative sample data. Secondly, upon imposing the learned prior, the inference problem requires global optimization of a given cost function. In this work, we will focus on binary-valued cost functions

$$E : \{0,1\}^n \rightarrow \mathbb{R} \tag{1}$$

over a large set of variables $\{x_1, \ldots, x_n\}$. The optimization of such functions has a long tradition, going back to the work of Ising on ferromagnetism [15]. Numerous methods have been proposed to tackle these combinatorial optimization problems. Geman and Geman [11] showed that the method of Simulated Annealing [16, 21] is guaranteed to find the global optimum of a given function. Alternative continuation methods such as Graduated Non-Convexity [4] have been proposed as well. Unfortunately, general purpose optimization methods such as Simulated Annealing require exponential runtime and can be quite slow for the number of nodes considered in most realistic applications.[1] In contrast, deterministic or approximation algorithms are not guaranteed to find a global optimum. The key challenge addressed in the present paper is therefore to devise methods to *efficiently* impose statistically learned knowledge in such combinatorial optimization problems.

The optimization of cost functions of the form (1) is in general an NP-hard combinatorial problem. The pioneering works of Picard and Ratliff [22] and of Greig *et al.* [13] showed that certain functions E of binary-valued variables can be represented by a directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with nonnegative edge weights and two nodes s and t, called source and sink, such that the optimum of the function E corresponds to the minimal s-t-cut of the respective graph. According to the theorem of Ford and Fulkerson [9], the computation of the minimal cut is equivalent to computing the maximum flow from the source to the sink. Several algorithms exist to compute this flow in polynomial time (see e.g. [5]). For applications of Graph Cuts to non-binary cases, we refer to [6, 14]. To restate, for certain combinatorial optimization problems, max-flow/min-cut algorithms provide both a fast and an exact solution.

Recently, theoretical efforts have been made to determine which classes of functions can be optimized by Graph Cuts. Ishikawa [14] provided constructive results showing how Graph Cuts may be applied to optimize Markov random fields for convex expressions. Kolmogorov and Zabih [17] pointed out that a class of energies satisfying certain submodularity constraints are *graph representable*, i.e. they can be efficiently minimized by computing the cut of an appropriate graph.

One should mention that Belief Propagation (BP) has become popular to efficiently perform Bayesian inference on graphs (see [10]). While BP is not limited by the above submodularity constraints, to the best of our knowledge there are no optimality guarantees for graphs with loops, such as the ones considered here.

The goal of the present paper is to provide a framework for learning empirical distributions of labels from sample graphs, to impose these as statistical priors in the framework of Bayesian inference on graphs and to specify which kinds of priors are consistent with graph-representable energy terms. The interpretation of submodularity in the context of statistical learning allows us to specify a class of priors which can be

[1] In practice, increased speed of Markov Chain Monte Carlo methods can be obtained by using bottom-up proposals and flipping entire patches of label values [2].

learned from samples and efficiently imposed within the framework of Bayesian inference. By restricting ourselves to graph-representable priors, we can guarantee global optima in polynomial time. In practice, we find the optimization times to be extremely fast.

As an illustration of our approach, we consider the problem of Bayesian restoration of binary images. In particular, we will show that one can impose previously learned information on correlation of the labels of pairs and triplets of vertices, as long as vertex labels are positively correlated. Numerical experiments demonstrate that fairly complex textural information can be learned, compactly represented and used for the efficient and optimal restoration from noisy images. While the restoration of binary textures may be considered a toy example, it shows that our method allows to impose statistically learned shape information in large-scale combinatorial optimization problems, providing global optima in polynomial runtime.

The outline of the paper is as follows. In Section 2, we will briefly review two lines of work which form the backbone of our method, namely the concept of Bayesian inference on graphs, and the submodularity conditions discussed in [17]. In Section 3, we introduce the key contribution of this paper, namely a characterization of a class of translation-invariant statistical priors on vertex labels which can be learned from sample graphs and which can be efficiently imposed in Bayesian inference via Graph Cuts. We define a measure of relevance of coupling terms which allows one to impose only the most relevant of learned priors. In Section 4, we provide numerical results on the restoration of binary images that illuminate different aspects of our method: highly accurate restorations despite large amounts of noise, optimal restorations of fairly complex textures in runtimes below one second, drastic speed-up through the use of sparse priors, and improved restoration by using higher-order priors.

2 Bayesian Inference on Graphs

Let $x = (x_1, \dots, x_n) \in \{0, 1\}^n$ be a vector of binary variables. Assume we are given a noisy version $I = (I_1, \dots, I_n) \in \mathbb{R}^n$ of this binary-valued vector. Then we can make use of the framework of Bayesian inference in order to reconstruct the vector x by maximizing the posterior probability

$$\mathcal{P}(x \mid I) = \frac{\mathcal{P}(I \mid x)\, \mathcal{P}(x)}{\mathcal{P}(I)}. \tag{2}$$

The Bayesian reasoning has become increasingly popular in the computer vision community [24], mainly for two reasons. Firstly, the conditional probability $\mathcal{P}(I \mid x)$ is often easier to model, since it represents the likelihood of a certain observation I given a state of the model x. Secondly, the Bayesian inference allows one to optimally integrate prior knowledge by the term $\mathcal{P}(x)$, specifying which interpretations of the data are *a priori* more or less likely.

In this paper, we will consider the specific case that the measurements I_i are mutually independent and that moreover they only depend on the value x_i at the node i. Under these assumptions, the data term in (2) can be written as: $\mathcal{P}(I \mid x) = \prod_i \mathcal{P}(I_i \mid x_i)$.

In this paper, we consider the data term:

$$P(I_i \,|\, x_i) \propto \exp\left(\frac{\lambda}{1 + |I_i - x_i|}\right). \tag{3}$$

While alternative choices are conceivable, this is not the focus of this work. The free parameter λ is currently chosen manually. Future research is focused on identifying an automatic estimate. The application of Bayesian inference amounts to a combinatorial optimization problem.

Kolmogorov and Zabih [17] recently discussed a class of cost functions which are able to be optimized efficiently by Graph Cuts. To this end, one considers two classes of cost functions denoted by $\mathcal{F}^2$ (and $\mathcal{F}^3$), representing functions E which can be written as a sum of functions of up to two variables at a time:

$$E(x_1, \ldots, x_n) = \sum_{i<j} E_{ij}(x_i, x_j), \tag{4}$$

and up to three variables for $\mathcal{F}_3$. In this way, one can consider nested classes of progressively more complex functions $\mathcal{F}^1 \subset \mathcal{F}^2 \subset \ldots \subset \mathcal{F}^n$, where the latter class corresponds to the full class of binary-valued functions.

In [17], Kolmogorov and Zabih pointed out that functions in $\mathcal{F}^1$, $\mathcal{F}^2$ and $\mathcal{F}^3$ can be optimized in polynomial time with the Graph Cuts algorithm if they fulfill certain *submodularity constraints* [18]. Namely, all functions in $\mathcal{F}^1$ are submodular, while functions in $\mathcal{F}^2$ and $\mathcal{F}^3$ are submodular if, for all terms $E_{ij}(x_i, x_j)$ of two arguments

$$E_{ij}(0,0) \;+\; E_{ij}(1,1) \;\leq\; E_{ij}(0,1) \;+\; E_{ij}(1,0), \tag{5}$$

and, for all terms $E_{ijk}(x_i, x_j, x_k)$ of three arguments, the same inequality must hold in the remaining two arguments once any one of them is fixed.

3 Statistical Priors for Bayesian Inference

In the context of restoration of binary images, researchers have successfully exploited generic priors $\mathcal{P}(x)$ on the space of label configurations x — such as the one used in the well-known Ising model [15] — which favor neighboring nodes to have the same label. Such priors lead to smooth restorations and are well suited for the removal of noise. Yet they also lead to a blurring of (possibly relevant) small-scale structures. Moreover, given sample images of the structures of interest, one may ask whether it is possible to *learn* more appropriate object-specific priors $\mathcal{P}(x)$ and impose these within the framework of Bayesian inference.

In this work, we are interested in priors which can be easily computed from the histograms of joint co-occurrence of label pairs or triplets, along the lines pioneered in [7, 12]. For a more sophisticated alternative to directly learn posterior distributions using MCMC sampling, we refer to [19]. To link statistical priors to co-occurrence frequencies, we rewrite the generic prior on a set of n variables as follows:

$$\begin{aligned}\mathcal{P}(x_1,\ldots,x_n) &= \mathcal{P}(x_1,x_2\,|\,x_3,\ldots,x_n)\mathcal{P}(x_3,\ldots,x_n)\\ &= \mathcal{P}(x_1,x_2\,|\,x_3,\ldots,x_n)\mathcal{P}(x_3,x_4\,|\,x_5,\ldots,x_n)\mathcal{P}(x_5,\ldots,x_n)\\ &= \ldots = \prod_{i\text{ odd}} \mathcal{P}(x_i,x_{i+1}\,|\,x_{i+2},\ldots,x_n).\end{aligned} \tag{6}$$

Let us now assume that the co-occurrence probability for any two variables does not depend on a third variable. Under this assumption, (6) then simplifies to

$$\mathcal{P}(x_1,\ldots,x_n) = \prod_{i\text{ odd}} \mathcal{P}(x_i,x_{i+1}). \tag{7}$$

Obviously, we can carry out the same rearrangement using arbitrary pairings of the n variables x_i. Upon multiplying all these equations, each pair (x_i,x_j) obviously appears the same number of times as a factor in the right-hand side. We get:

$$\left(\mathcal{P}(x_1,\ldots,x_n)\right)^{\Gamma} = \prod_{i\neq j} \mathcal{P}(x_i,x_j), \tag{8}$$

where the constant Γ denotes the number of ways to generate such pairings divided by the number of times each pair appears in the overall product. In the case of label pairs, we have $\Gamma = \binom{n}{2}$. We obtain the *prior energy*:

$$E(x_1,\ldots,x_n) = -\log\mathcal{P}(x_1,\ldots,x_n) = -\frac{1}{\Gamma}\sum_{i\neq j}\log\mathcal{P}(x_i,x_j). \tag{9}$$

Similarly, the relaxed assumption that the co-occurrence of labels for any triplet (x_i,x_j,x_k) does not depend on a fourth node, leads to an energy of the form

$$E(x_1,\ldots,x_n) = -\frac{1}{\tilde{\Gamma}}\sum_{ijk}\log\mathcal{P}(x_i,x_j,x_k), \tag{10}$$

where the sum extends over all pairwise distinct triplets of nodes and $\tilde{\Gamma} = \binom{n}{3}$. While the above independency assumptions will generally not be fulfilled, let us make two remarks: Firstly, the expressions for the priors (9) and (10) also hold if higher-order effects do not contribute *on the average*. Secondly, the independency assumption can be gradually relaxed by considering terms of increasing order of interaction. We will refer to priors with an energy $E \in \mathcal{F}^k$ as priors of order k. In the following, we will focus on the spaces $\mathcal{F}^2$ and $\mathcal{F}^3$. To circumvent the approximation in (7), the Markov random field community has developed more sophisticated techniques to approximate the prior in terms of local characteristics (see e.g. [24]).

For a second-order prior $\mathcal{P}$, the energy E in (6) is of the form (4). Since we are dealing with binary-valued variables, the each term E_{ij} in (4) is of the form

$$E_{ij}(x_i,x_j) = \alpha_{ij}^{11}x_ix_j + \alpha_{ij}^{10}x_i(1-x_j) + \alpha_{ij}^{01}(1-x_i)x_j + \alpha_{ij}^{00}(1-x_i)(1-x_j), \tag{11}$$

with four parameters associated with each vertex pair. According to (6), we can relate these parameters to the probability of co-occurrence of label values:

$$\alpha_{ij}^{11} = -\log\mathcal{P}\left(x_i\!=\!1\cap x_j\!=\!1\right), \quad \alpha_{ij}^{10} = -\log\mathcal{P}\left(x_i\!=\!1\cap x_j\!=\!0\right), \;\ldots \tag{12}$$

In the case of a third-order prior on binary-valued variables, the energy E in (6) is given by a sum of energies E_{ijk} taking on the form

$$E_{ijk}(x_i, x_j, x_k) = \alpha^{111}_{ijk} x_i x_j x_k + \alpha^{110}_{ijk} x_i x_j (1 - x_k) + \alpha^{101}_{ijk} x_i (1 - x_j) x_k + \dots$$

with eight parameters associated with each vertex triplet and

$$\alpha^{111}_{ijk} = -\log \mathcal{P}(x_i = 1 \cap x_j = 1 \cap x_k = 1), \quad \alpha^{110}_{ijk} = \dots \tag{13}$$

The central idea of learning priors is to determine the parameters of the probabilistic model (6) from samples of labeled graphs. According to (13), the parameter α^{111}_{ijk}, for example, corresponds to the negative logarithm of the relative frequency of label configuration $(1, 1, 1)$ at the three nodes i, j and k.

In most relevant restoration algorithms one does not know the location of structures of interest. Therefore it is meaningful to focus on the subclass of *translation-invariant priors*, i.e. priors which treat all nodes identically. These are also referred to as *spatially homogeneous priors* [24]. For priors of second order, the model parameters in expression (11) can only depend on the *relative* location of node i and node j. In other words $\alpha_{ij} = \alpha_{(j-i)}$ etc., where $(j - i)$ denotes the vector connecting node i to node j. Given a training image, one can estimate the parameters $\alpha^{11}_{(j-i)}, \alpha^{01}_{(j-i)}, \alpha^{10}_{(j-i)}$, and $\alpha^{00}_{(j-i)}$ defining the translation-invariant prior distributions of second order, because the probabilities of co-occurrence of label pairs in (12) can be approximated by their histogram values. Similarly, in the case of third-order priors, the eight parameters α_{ijk} in (3) associated with each triplet of nodes only depend on the relative location of nodes i, j and k. These parameters can be estimated from joint histograms of triplets computed on a sample image.

Along the lines sketched above, it is possible to learn priors on the set of binary variables from the empirical histograms computed on sample images. Such statistical priors can be used in various ways. For example, as suggested by Shannon, one could generate synthetic label configurations (binary images if the nodes correspond to image pixels) by randomly sampling from the estimated distributions — see for example [7]. In the following, we will instead employ the empirically learned priors for the purpose of reconstructing a labeling $x = \{x_1, \dots, x_n\} \in \{0, 1\}^n$ of a graph given a noisy version $I = \{I_1, \dots, I_n\} \in \mathbb{R}^n$ of it and given the knowledge that the labeling is statistically similar to previously observed label configurations. The optimal restoration is given by the maximum *a posteriori* estimate in (2). Equivalently, we can minimize the negative logarithm of (2). With (3) and a translation-invariant prior of second order obtained from equations (9), (4) and (11) this leads to an energy of the form:

$$\begin{aligned} E(x_1, .., x_n) = \sum_i \frac{-\lambda}{1 + |I_i - x_i|} + \sum_{i<j} \Big(& \alpha^{11}_{(j-i)} x_i x_j + \alpha^{10}_{(j-i)} x_i (1 - x_j) \\ & + \alpha^{01}_{(j-i)} (1 - x_i) x_j + \alpha^{00}_{(j-i)} (1 - x_i)(1 - x_j) \Big). \end{aligned} \tag{14}$$

Similarly binary restoration with a translation-invariant prior of third order is done by minimizing an energy of the form:

$$E(\{x_i\}) = \sum_i \frac{-\lambda}{1+|I_i - x_i|} + \sum_{i<j<k} \Big(\alpha^{111}_{(j–i,k–i)} x_i x_j x_k + \alpha^{110}_{(j–i,k–i)} x_i x_j (1-x_k) + ...\Big), \tag{15}$$

with eight terms imposing learned correlations of the label at node i with labels at nodes j and k. Due to the translation invariance, the parameters $\alpha_{ijk} = \alpha_{(j–i,k–i)}$ merely depend on the vectors from i to j and from i to k.

Minimizing energies of the forms (14) or (15) over the space of binary variables $x \in \{0,1\}^n$ is in general a hard combinatorial problem.[2] In the context of images with relevant size, the number of nodes is on the order of $n \sim 256^2$ or larger, therefore an exhaustive search or stochastic optimization methods such as simulated annealing are not well-suited for this task.

While the Graph Cuts algorithm allows an efficient global optimization in polynomial time, it only applies to a certain class of energies. The submodularity constraints reviewed in Section 2, however, allow us to make a precise statement about which priors *can* be efficiently imposed in the Bayesian restoration using Graph Cuts. Using the relation between energies and prior distributions given in (9), we can express the submodularity constraint (5) in terms of probabilities:

$$-\log \mathcal{P}_{00} - \log \mathcal{P}_{11} \leq -\log \mathcal{P}_{01} - \log \mathcal{P}_{10}, \tag{16}$$

where $\mathcal{P}_{00} = \mathcal{P}(x_i = 0 \cap x_j = 0)$ stands for the probability that both labels are 0 etc. The above inequality is equivalent to the requirement that:

$$\mathcal{P}_{00}\, \mathcal{P}_{11} \geq \mathcal{P}_{01}\, \mathcal{P}_{10}. \tag{17}$$

If the joint probability of label values at nodes i and j fulfills the above inequality, then it can be efficiently imposed in the Bayesian restoration by solving the respective max-flow/min-cut problem. In particular, this implies that for any two nodes which are positively correlated (i.e. $\mathcal{P}_{00} \geq \max\{\mathcal{P}_{01}, \mathcal{P}_{10}\}$ and $\mathcal{P}_{11} \geq \max\{\mathcal{P}_{01}, \mathcal{P}_{10}\}$), one can impose their joint probability within the Graph Cuts framework. Beyond this, one can also integrate priors stating that, for example, the label configuration (01) dominates all other configurations while the configuration (10) is sufficiently unlikely for inequality (17) to be fulfilled. On the other hand, joint priors modeling negative correlation, where opposite labels (01) and (10) dominate, are not consistent with inequality (17).

Similarly, the submodularity constraints in [17] impose conditions for which the distributions of triplets can be imposed within the Graph Cuts optimization. Namely, the inequalities have to hold with respect to the remaining two arguments once any one of them is fixed, i.e. if $x_i = 0$ is fixed then the inequality in nodes j and k states:

$$\mathcal{P}_{000}\, \mathcal{P}_{011} \geq \mathcal{P}_{001}\, \mathcal{P}_{010}, \tag{18}$$

where $\mathcal{P}_{000} = \mathcal{P}(x_i = 0 \cap x_j = 0 \cap x_k = 0)$ represents the joint occurrence of three labels of 0, etc. There are eight such constraints for each triplet.

In practice, we compute these joint histograms from sample images and retain only those priors which are consistent with the submodularity constraints (17) or (18). The

[2] For an example of an NP-hard problem in the class $\mathcal{F}^2$ see [17].

resulting cost function can be efficiently optimized by the Graph Cuts algorithm. In other words: once we have selected an appropriate set of statistically learned priors, we can perform the Bayesian inference in polynomial runtime. For details on how to convert energy terms into respective edge weights of a graph, we refer to [17].

While the global optimum of the resulting restoration problem is guaranteed to be computable in polynomial time, experimental evidence shows that increasing the number of constraints (and thereby the number of edges in the graph) will typically increase the computation time: While the computation time for $n = 256^2$ nodes with four constraints per node was on the order of 0.03 seconds, increasing the number of constraints per node to 716 leads to a computation time of more than one minute. A simple remedy to this problem is to only impose the most *relevant* constraints. The submodularity constraint in (5) guarantees that the edges of the corresponding graph have non-negative weights [17]. Moreover, if the left side of inequality (5) is much smaller than the right side, then the respective edges will have very large positive weights, hence they will be very relevant to the computation of the minimal cut. Therefore, we can heuristically define the relevance of a coupling term (11) between nodes i and j as the weight of introduced edges:

$$\mathrm{rel}_{ij} = \alpha_{ij}^{10} + \alpha_{ij}^{01} - \alpha_{ij}^{11} - \alpha_{ij}^{00}. \tag{19}$$

In the context of priors of third order, there are six submodularity constraints associated with each node triplet. As a measure of the relevance of a given triplet of nodes, we simply compute the mean of the associated six relevance measures in (19). Qualitatively, this relevance measure states that the co-occurrence of identical label values should dominate the histogram for a prior to be relevant.

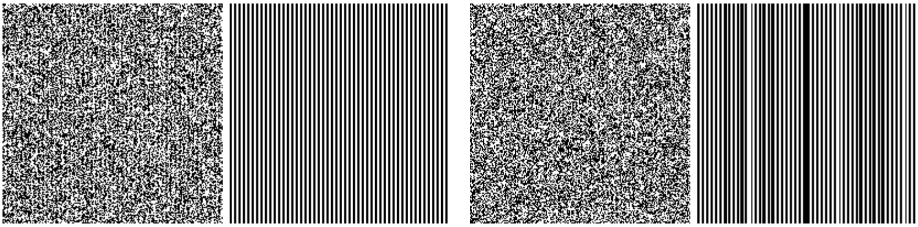
4 Experimental Results

Figure 1 shows a binary pattern of vertical stripes of width two pixels, corrupted by various amounts of salt-and-pepper noise.[3] The second image shows the restoration (with $\lambda = 1$) obtained using a second order prior coupling each pixel to the two nodes directly above and to the right. The priors estimated from empirical histograms of stripe patterns simply state that vertically neighboring pixels are very likely to be of the same color. There is no preference in the horizontal direction: since the stripes are two pixels wide, all pair combinations are equally likely. As a consequence, the restoration of the noisier version is suboptimal in that the vertical stripes in the restoration are no longer equidistantly spaced.[4] With increasing noise level, the Bayesian restoration requires increasingly sophisticated priors. The above prior on neighboring pairs of labels can be extended in two ways: by increasing the neighborhood size and by generalizing to higher-order interactions.

By increasing the neighborhood window in which priors are learned and imposed, the resulting prior is still of second order, but it integrates correlations of a given node with more distant nodes. In the case of the stripe pattern in Figure 1, we learned the joint probabilities for a pixel and its neighbors in 9×9 window. This provides coupling

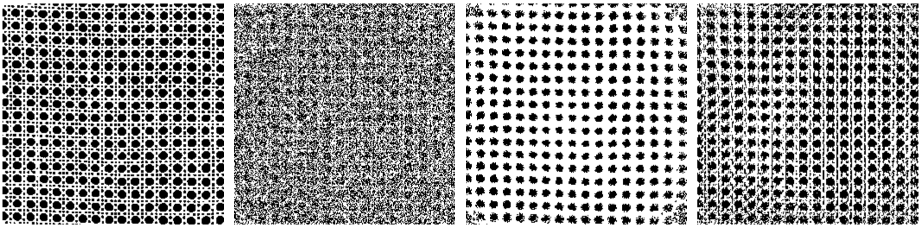
[3] "80% noise" means 80% of the pixels are replaced by a random value.

[4] The restoration error gives the percentage of incorrectly labeled pixels.



80% noise Restor. (0% error) 90% noise Restor. (7% error)

Fig. 1. Fast restoration of simple patterns: Optimal restorations of noisy stripe patterns using statistical priors learned from the joint histograms of a pixel with the neighbor above and the neighbor to the right. While the left image was perfectly restored in 0.02 seconds, the right one has a restoration error[6] of 7% in 0.03 seconds (on a 200 × 200 image). Including couplings in larger neighborhoods improves the restoration.



Brodatz texture with 70% noise Restor. (375 constr.) Restor. (5 constr.)

Fig. 2. Efficient restoration of complex textures: The images on the left show a binarized Brodatz texture with 70% of noise. Using only relevant constraints (right image), the algorithm is not only faster, but it also provides a better restoration. See Table 1 for a numerical comparison.

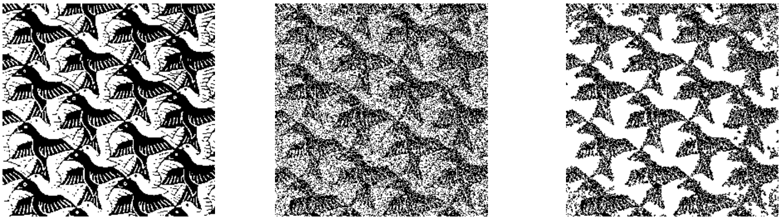
to 40 neighbors, 22 of which are submodular. This prior allows to identify horizontal correlations. In the case of the stripe pattern in Figure 1, bottom, it provides a perfect restoration in 1.6 seconds for an image of size 200 × 200, with $\lambda = 1$.[5]

In order to restore more complex patterns, it is necessary to consider joint distributions of labels in increasingly large neighborhoods. This will lead to an increasing number of edges in the respective graph, coupling each pixel to a larger and larger number of surrounding pixels. In order to keep the computation time low, we impose only the most relevant constraints according to the measure defined in (19). Figure 2 shows a binarized Brodatz texture (256 × 256 pixels) and the same texture with 70% salt-and-pepper noise. On a sample texture image, we estimated the pairwise joint distributions for pixel couplings in a neighborhood of 35 × 35 pixels. Among these 612 possible neighbor nodes, 375 provided submodular constraints fulfilling the inequality (17). Using all 375 constraints, the computation of the optimal restoration took 23.2 seconds, giving a restoration error of 23.6%. Using only the five most relevant constraints allowed an optimal restoration in 0.4 seconds. Surprisingly, the restoration error was only 20%. Respective restorations are shown in Figure 2, third and fourth image.

[5] Imposing pair priors on a neighborhood size of 9 × 9, we found that one obtains perfect restorations of the stripe pattern in Figure 1 even with 99% noise.

Table 1. Efficiency with sparse priors: Run time, restoration error and appropriate λ values for decreasing number of constraints imposed in the restoration of the Brodatz texture (Fig. 2). Using only the most relevant constraints leads to improvements both with respect to the run time and, surprisingly, with respect to the restoration error (up to a minimal set of constraints) — see text. The highlighted error values are associated with the restorations in Fig. 2.

Number of constraints	375	53	21	13	7	5	3
CPU time (s)	23.2	2.92	1.45	0.86	0.47	0.40	0.33
Restoration error (%)	**23.6**	23.6	22.2	21.2	20.0	**20.0**	23.3
λ	38	38	33	20	13	8	4



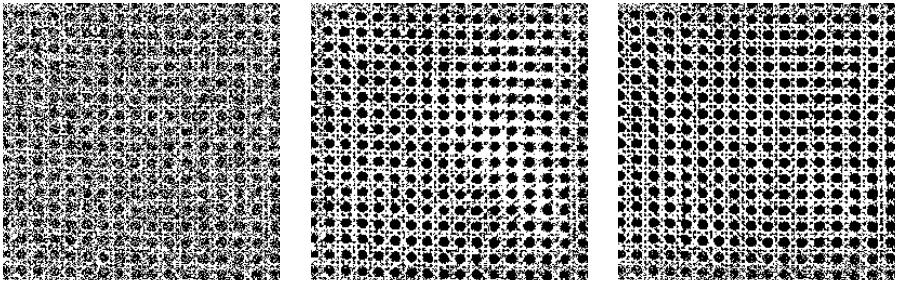
Escher drawing with 50% noise Restoration (19.6% error)

Fig. 3. Larger neighborhood systems: Restoration of a noisy drawing of M. C. Escher using the 20 most relevant second order constraints estimated in a 130×130 window. In contrast to generic smoothness priors, the statistically learned priors do not lead to a blurring of image structures.

Table 1 shows respective run-times, restoration errors and appropriate values of λ for imposing varying numbers of relevant constraints which were selected by thresholding the relevance (19) computed for each node pair. The computation time decreases with fewer constraints used. Moreover, the restoration error decreases when using only the most relevant constraints (up to a certain minimal set of constraints). We believe that this property is due to the fact that less relevant constraints may impose spurious correlations, especially when computed from not perfectly periodic textures such as the Brodatz texture. Using only the relevant constraints will assure that the algorithm makes use of only those couplings that are persistent throughout the entire texture.

The selection of relevant terms becomes more crucial when learning priors for larger-scale structures, as these require to consider larger neighborhoods. Figure 3 shows the restoration of a noisy version of a drawing by M. C. Escher.

As suggested in Section 2, one can learn and impose priors on the joint distribution of triplets of labels — provided that the submodularity conditions (18) are fulfilled. In practice, the key difficulty of learning third-order priors is that the consideration of all possible node triplets is infeasible for graphs of meaningful size: For a graph of 256×256 nodes, there exist $\binom{256^2}{3} \approx 5 \cdot 10^{13}$ possible triplets. To consider all possible triplets within a certain neighborhood of each node (without counting some more often than others) turns out to be a challenging problem as well. In order to count all triplets in



Noisy Brodatz texture 2nd order restor. ($\epsilon = 14.4\%$) 3rd order restor. ($\epsilon = 13.5\%$)

Fig. 4. Triplets versus pairs: Restoration using priors of second and third order on a Brodatz texture with 50% noise. Both priors impose the eleven most relevant constraints in a neighborhood of 15 pixels. Including terms of third order reduces the reconstruction error ϵ from 14.4% (computed in 0.5 seconds) to 13.5% (computed in 2.8 seconds). Exploiting knowledge about the joint probability of triplets (rather than pairs) provides additional submodularity of the reconstruction.

a certain "vicinity" of a node, we revert to the following solution: For each node of the graph, we consider all triangles of a fixed maximal circumference δ (measured in the Manhattan distance) with one vertex at the node of interest. The parameter δ provides a measure of the "vicinity" analogous to the window size in the case of pairs. Figure 4 shows restorations of a noisy Brodatz texture obtained with second and third order priors, respectively. In the specified neighborhood, we identified 215760 triplets per node, 7873 of which provided submodular constraints. We used a threshold $\theta = 2.1$ on the respective relevance of pairs (or triplets) — see (19) — leaving eleven constraints for each node in the graph. Imposing constraints on the joint distribution of triplets (rather than pairs) reduced the restoration error ϵ from 14.4% to 13.5%.

5 Conclusion

We proposed to introduce statistically learned priors into an efficient method for Bayesian inference on graphs. Building up on submodularity constraints for graph-representability, we specified a class of spatially homogeneous priors of second and third order which can be learned from co-occurrence histograms and which can be efficiently imposed by computing Graph Cuts. In particular, we showed that priors favoring labels to be similar are part of this class. To the best of our knowledge, this is the first time that statistically learned priors of second and third order were introduced into an efficient and exact combinatorial optimization algorithm. We believe that our contribution will help to bridge the gap between statistical learning for Bayesian inference and efficient combinatorial optimization. As an illustration of our method, we demonstrated that one can compute optimal restorations of rather complex binary textures from images which are heavily corrupted by noise in runtimes on the order of seconds. Future work aims at answering several open questions: Are there graph-representable priors beyond the class considered here? Are there ways of generalizing the invariance group from translation to rotation and scale invariance?

References

1. K. Abend, T. Harley, and L. N. Kanal. Classification of binary random patterns. *IEEE Transactions on Information Theory*, 11:538–544, 1965.
2. A. Barbu and S.-C. Zhu. Generalizing Swendsen-Wang to sampling arbitrary posterior probabilities. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 27(8):1239–1253, 2005.
3. J. Besag. On the statistical analysis of dirty pictures. *J. Roy. Statist. Soc., Ser. B.*, 48(3):259–302, 1986.
4. A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, 1987.
5. Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 26(9):1124–1137, 2004.
6. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 23(11):1222–1239, 2001.
7. G.R. Cross and A.K. Jain. Markov random fields texture models. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 5(1):25–39, 1983.
8. H. Derin and H. Elliott. Modeling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 9(1):39–55, Jan. 1987.
9. L. Ford and D. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton, New Jersey, 1962.
10. W.T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *Int. J. of Computer Vision*, 40(1):24–57, 2000.
11. S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 6(6):721–741, 1984.
12. G. Gimelfarb. Texture modeling by multiple pairwise pixel interaction. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 18(11):1110–1114, 1993.
13. D. M. Greig, B. T. Porteous, and A. H. Seheult. Exact maximum *a posteriori* estimation for binary images. *J. Roy. Statist. Soc., Ser. B.*, 51(2):271–279, 1989.
14. H. Ishikawa. Exact optimization for Markov random fields with convex priors. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 25(10):1333–1336, Oct. 2003.
15. E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, 23:253–258, 1925.
16. S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.
17. V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 24(5):657–673, 2004.
18. B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer, 3rd edition, 2006.
19. S. Kumar and M. Hebert. Approximate parameter learning in discriminative fields. In *Snowbird Learning Workshop*, Utah, 2004.
20. B. S. Manjunath and R. Chellappa. Unsupervised texture segmentation using Markov random field models. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 13(5):478–482, May 1991.
21. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Physics*, 21:1087–1092, 1953.
22. J. C. Picard and H. D. Ratliff. Minimum cuts and related problems. *Networks*, 5:357–370, 1975.
23. W. Pieczynski, D. Benboudjema, and P. Lanchantin. Statistical image segmentation using triplet Markov fields. In Sebastiano B. Serpico, editor, *SPIE Int. Symposium on Image and Signal Processing for Remote Sensing VIII*, volume 4885, pages 92–101. SPIE, March 2003.
24. G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*, volume 27 of *Appl. of Mathematics*. Springer, Heidelberg, 2003.