IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 10, pp. 1333-1336. October 2003.

Exact optimization for Markov random fields with convex priors

Hiroshi Ishikawa

Courant Institute of Mathematical Sciences

New York University

251 Mercer Street, New York, NY 10012

http://cs.nyu.edu/ishikawa/

Abstract

We introduce a method to solve exactly a first order Markov Random Field optimization problem in more generality than was previously possible. The MRF shall have a prior term that is convex in terms of a linearly ordered label set. The method maps the problem into a minimum-cut problem for a directed graph, for which a globally optimal solution can be found in polynomial time. The convexity of the prior function in the energy is shown to be necessary and sufficient for the applicability of the method.

To the memory of Dr. Henning Biermann (*October 16th, 1972 – July 1st, 2002*.)

1 Introduction

It is rarely possible to solve exactly a large combinatorial optimization problem. Yet, there are exceptional circumstances where one can use a known method to find a global optimum. For instance, in a situation where the state of the problem is described as a string of linearly ordered local states, dynamic programming can be used (e.g. Amini, Weymouth, and Jain[1]; Baker and Binford[2]; Geiger, Gupta, Costa, and Vlontzos[6]; Montanari[17]). This paper points out another such instance and describes a method that can be used; namely, a method to solve a first order Markov Random Field (MRF) with a prior term that is convex in terms of a linearly ordered label set. The definition of an MRF is given in the next section.

Simulated annealing has been used to solve certain MRF problems, although it is also notoriously slow. It is a kind of stochastic optimization: while gradient descent methods go straight for the nearest minimum as fast as possible, always going downhill and never going uphill, stochastic optimization algorithms randomly allow occasional uphill jumps, enabling escape from local minima and the possibility of finding the global minimum (Metropolis et al[16]). The simulated annealing algorithm initially allows such jumps with a high probability; then, according to some "annealing schedule", it gradually lets the probability decrease to zero. The probability is usually parametrized by one parameter called temperature. Geman and Geman[7] popularized the MRF in the vision/image processing community. They use simulated annealing for image restoration by Maximum A Posteriori (MAP) estimation, and prove the "annealing theorem," which says roughly that if the temperature decreases like $(\log(1+t))^{-1}$ with time t, the algorithm is guaranteed to reach the MAP solution, although it takes an infinite time to achieve the solution.

The method we describe here is a generalization of the method by Greig, Porteous, and Seheult[8, 9]. Their method guarantees a global optimum in solving a first order MRF with two label states. They compare the results of their method with simulated annealing in binary image restoration. Although simulated annealing is also theoretically guaranteed to reach the optimal solution eventually, they found that in practice simulated annealing tended to over-smooth the noisy image. Here, we generalize their method to MRF's with three or more linearly ordered states and convex priors, using a minimum-cut algorithm on a directed graph, and also give a precise criterion on the form of the problem for the method to be applicable. Roy and Cox[19] also use a maximum-flow algorithm in a similar way to solve stereo problems without epipolar lines. (Roy[18] later clarified their method from the point of view of linear energy minimization.) Boykov, Veksler, and Zabih[4] give an approximate solution for MRF problems with more than two labels and semi-metric prior functions. We have used our method in image restoration, segmentation, and stereo[10, 11, 12, 13, 14].

In the next section, we discuss the MRF formulation in more detail and state precisely the class of problems the proposed method can solve. Then in Section 3, we describe the method itself, then prove that the method solves exactly the class of problems specified.

2 Markov Random Fields

A graph $G = (V, E)$ consists of a finite set V of vertices and a set $E \subset V \times V$ of edges. An edge $(u, v) \in E$ is said to be from vertex u to vertex v. An undirected graph is one in which all edges go both ways: $(u, v) \in E$ iff $(v, u) \in E$. A clique is a set of vertices in an undirected graph in which every vertex has an edge to every other vertex.

An MRF consists of an undirected graph $G = (V, E)$ without loop edges (i.e., edges of the form (v, v)), a finite set $L = \{l_1, \ldots, l_k\}$ of labels, and a probability distribution P on the space $\mathcal{X} = L^V$ of label assignments. That is, an element X of $\mathcal{X}$, sometimes called a configuration of the MRF, is a map that assigns to each vertex v a label X_v in L. Let $\mathcal{N}_v$ denote the set of neighbors $\{u \in V \,|\, (u, v) \in E\}$ of vertex v. Also, for a configuration $X \in \mathcal{X}$ and $S \subset V$, let X_S denote the event $\{Y \in \mathcal{X} \,|\, Y_v = X_v, \forall v \in S\}$. By definition, the probability distribution must satisfy the condition:

$$P(X) > 0 \text{ for all } X \in \mathcal{X}$$
$$P(X_{\{v\}}|X_{V\setminus\{v\}}) = P(X_{\{v\}}|X_{\mathcal{N}_v}).$$

This condition states that the assignment of a label to a vertex is conditionally dependent on the assignment to other vertices only through its neighbors.

Note that the MRF is a conditional probability model. A theorem(Besag[3], Kinderman and Snell[15]) connects it to a joint probability model: a probability distribution P on $\mathcal{X}$ is an MRF exactly when it is a Gibbs distribution relative to G:

$$P(X) \sim \mathrm{e}^{-E(X)},$$
$$E(X) = \sum_{C\in\mathcal{C}} E_C(X),$$

where $\mathcal{C}$ denotes the set of cliques in G and E_C a function on $\mathcal{X}$ with the property that $E_C(X)$ depends only on values of X on C.

In computer vision and image processing, many problems can be put in terms of MRF. We often wish to find a configuration with the minimum energy, or solve the MRF problem. Note that in the above definition no data or observation, such as the image, appears. Any data or observation that affects the statistics is implicit in the probability distribution.

The simplest interesting case is when only the edges and vertices, the two simplest kinds of cliques, influence the potential:

$$E(X) = \sum_{(u,v)\in E} g(u,v,X_u,X_v) + \sum_{v\in V} h(v,X_v). \tag{1}$$

This is called a first order MRF. The first term in (1) is sometimes called the prior, as it often derives from the prior model in a MAP optimization problem. What is solved in [8, 9] is the case with a label set $L = \{0,1\}$ and the first order energy (1) with $g(u,v,X_u,X_v) = \alpha_{uv}(1-\delta_{X_u,X_v})$, where $\alpha_{uv} \geq 0, \alpha_{uv} = \alpha_{vu}$ and $\delta_{X,Y}$ gives 1 if $X = Y$ and 0 otherwise.

In this paper, we add to the class of first order MRF problems that can be exactly solved those with more than two labels and energies of the form:

$$E(X) = \sum_{(u,v)\in E} \alpha_{uv}\, g(\iota(X_u) - \iota(X_v)) + \sum_{v\in V} h(v, X_v), \tag{2}$$

where $\alpha_{uv} \geq 0, \alpha_{uv} = \alpha_{vu}$, and $g(x)$ is a convex function. The function ι gives the index of a label:

$$\iota(l_i) = i.$$

Although ι might seem a strange function, it just expresses that there is a linear order among the labels and that the prior depends only on the difference of their ordinal numbers.

By definition, a real-valued function $g(x)$ on the real numbers is convex if $g(ax + (1 - a)y) \leq ag(x) + (1 - a)g(y)$ for all x, y, and $0 \leq a \leq 1$. When we deal with a function on a discrete set, this condition holds for a convex function $g(x)$ whenever x, y and $ax + (1 - a)y$ are in the set. It is natural to make it into the definition of convexity on a discrete set:

Definition. *A real-valued function $g(x)$ on a set A of real numbers is convex if*

$$g(ax + (1 - a)y) \leq ag(x) + (1 - a)g(y)$$

holds for any $x, y \in A$ and $0 \leq a \leq 1$ such that $ax + (1 - a)y \in A$.

Note that the convexity of a function depends on the set it is defined. It is easy to see that for a function $g(x)$ on a set of consecutive integers to be convex, it is necessary and sufficient that all second differences are nonnegative:

$$g(x + 1) - 2g(x) + g(x - 1) \geq 0.$$

Coming back to our energy functional, we note here as a particular case that if $L = \{l_1, \ldots, l_k\}$ is a set of evenly spaced real numbers in ascending (or descending) order and $\hat{g}(x)$ is a convex function, an MRF with energy

$$E(X) = \sum_{(u,v)\in E} \alpha_{uv}\, \hat{g}(X_u - X_v) + \sum_{v\in V} h(v, X_v)$$

satisfies the condition, as ι is an affine function in that case and thus the function $g(x)$ defined by $g(\iota(l) - \iota(l')) = \hat{g}(l - l')$ is convex if and only if $\hat{g}(x)$ is.

Without loss of generality, we can assume the labels to be consecutive integers $L = \{1, \ldots, k\}$ so that ι becomes an identity, since we can always redefine $h(v, l)$ by $h(v, \iota^{-1}(l))$. Thus the energy (2) becomes:

$$E(X) = \sum_{(u,v)\in E} \alpha_{uv}\, g(X_u - X_v) + \sum_{v\in V} h(v, X_v). \tag{3}$$

In this paper, we propose a method that uses a minimum cut algorithm to solve exactly this class of MRF problems.

3 Solving MRF by Minimum Cut

3.1 Maximum Flow and Minimum Cut

Here, we remind the reader of the formulation of the maximum flow and minimum cut problem.

The maximum flow problem and its dual, the minimum cut problem, are classical combinatorial problems with a wide variety of scientific and engineering applications. The maximum flow problem and related flow and cut problems have been studied intensively for over three decades and are standard in textbooks (e.g., [5]).

Consider a graph $G = (V, E)$ with a function c on $V \times V$ such that $c(u, v) = 0$ if $(u, v) \notin E$. We call the function the capacity function. Choose two special vertices, a source s and a sink t. A flow with respect to the triple (G, s, t) is a function $f : V \times V \to \mathbf{R}$ such that:

1. $f(u, v) \le c(u, v)$ for all $u, v \in V$.

2. $f(u, v) = -f(v, u)$ for all $u, v \in V$.

3. $\sum_{v\in V} f(u, v) = 0$ for all $u \in V \setminus \{s, t\}$.

The value $|f|$ of a flow f is defined as

$$|f| = \sum_{v \in V} f(s, v).$$

A maximum flow is a flow with the maximum value.

For the same triple (G, s, t), a cut is a partition of V into two subsets S and $T = V \setminus S$ such that $s \in S$ and $t \in T$. For a given cut, the *total cost* of the cut is defined as

$$\sum_{u \in S, v \in T} c(u, v).$$

When an edge has its tail in S and head in T, the edge is said to be *in the* cut. A minimum cut is a cut with the minimum total cost.

The max-flow min-cut theorem says that by finding a maximum flow, a minimum cut can be found. When the capacity function always takes nonnegative values, a maximum flow, therefore a minimum cut, can be found in polynomial time using several known algorithms.

In the following subsections, we describe the method to efficiently obtain global optimum for any MRF problem with this form of energy.

3.2 The Graph

The main idea of the method is to define a graph such that

1. there is a one-to-one correspondence between configurations of the MRF and cuts of the graph, and

2. the total cost of the cut is exactly the same as the total energy of the configuration.

With such a graph, we can find a minimum energy configuration of the MRF by finding a minimum cut of the graph.

As we mentioned above, without loss of generality we can assume the labels to be the consecutive integers: $L = \{1, \ldots, k\}$. Also, we can assume that the function $h(v, l)$ takes nonnegative values, since otherwise we can always redefine it by taking its minimum value

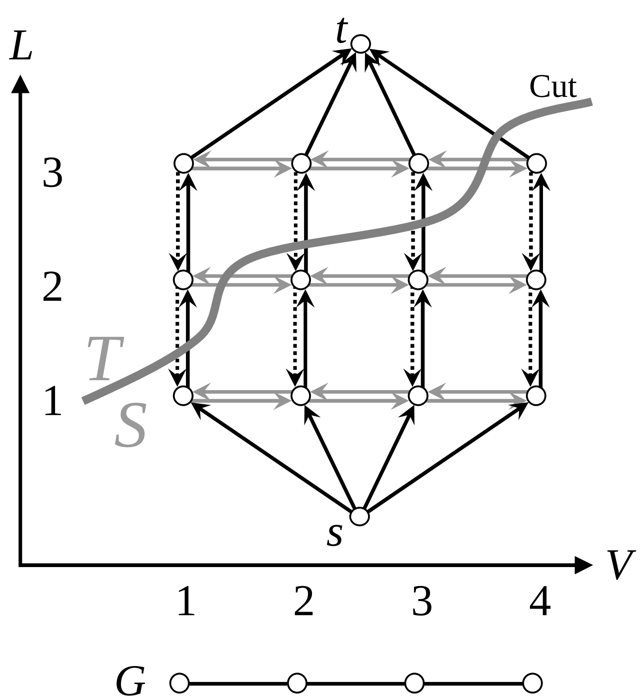

Figure 1: Data edges are depicted as black arrows. Four of them are in the cut here, representing the assignments $X_1 = 1$, $X_2 = 2$, $X_3 = 2$, and $X_4 = 3$. Penalty edges are represented by horizontal arrows. By crossing consecutive penalty capacities, the cost is added linearly, realizing the prior $g(x) = |x|$. With more edges, any convex $g(x)$ can be used. Constraint edges are depicted as dotted arrows. They ensure that the assignment X_v is uniquely determined for each v. These edges cannot be in the cut, and thus they prevent the cut from "going back".

over all possible pairs of $v \in V$ and $l \in L$ and subtract it from the function without changing the optimization problem. Define a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ as follows.

$$\begin{aligned} \mathcal{V} &= V \times L \cup \{s,t\} = \{u_{w,i} \mid w \in V;\ i \in L\} \cup \{s,t\}, \\ \mathcal{E} &= \mathcal{E}_{\mathrm{D}} \cup \mathcal{E}_{\mathrm{C}} \cup \mathcal{E}_{\mathrm{P}}. \end{aligned} \tag{4}$$

Below, each of the three subsets of edges $\mathcal{E}_{\mathrm{D}}, \mathcal{E}_{\mathrm{C}}, \mathcal{E}_{\mathrm{P}}$ is defined and their capacities are specified. The reader is referred to Figure 1 for illustration.

3.3 Data Edges

Data edges implement the data term $h(v, X_v)$ in the energy. They are shown in Figure 1 as black arrows going up. The set of data edges is defined by:

$$\begin{aligned} \mathcal{E}_{\mathrm{D}} &= \bigcup_{v \in V} \mathcal{E}_{\mathrm{D}}^v, \\ \mathcal{E}_{\mathrm{D}}^v &= \{(s, u_{v,1})\} \cup \{(u_{v,i}, u_{v,i+1}) | i = 1, \ldots, k-1\} \cup \{(u_{v,k}, t)\}. \end{aligned} \tag{5}$$

For each vertex v of the original graph G, $\mathcal{E}_{\mathrm{D}}^v$ is the series of edges $s \to u_{v,1} \to u_{v,2} \to \cdots \to u_{v,k} \to t$, which we call the column over v.

The capacities of these edges are defined by

$$\begin{aligned} c(s, u_{v,1}) &= +\infty, \\ c(u_{v,i}, u_{v,i+1}) &= h(v, i); \quad i = 1, \ldots, k-1, \\ c(u_{v,k}, t) &= h(v, k). \end{aligned} \tag{6}$$

These capacities are defined so that the sum of the capacities of data edges in the cut equals the data term $h(v, X_v)$ in the energy, according to the one-to-one correspondence between configurations of the MRF and cuts of the graph, which we define next.

3.4 Constraint Edges

Since any cut of the triple $(\mathcal{G}, s, t)$ separates s and t, at least one data edge in the column $\mathcal{E}_{\mathrm{D}}^v$ over each $v \in G$ is in the cut. Constraint edges guarantee that each column is cut exactly once. They are the edges opposite to data edges:

$$\begin{aligned} \mathcal{E}_{\mathrm{C}} &= \bigcup_{v \in V} \mathcal{E}_{\mathrm{C}}^v, \\ \mathcal{E}_{\mathrm{C}}^v &= \{(u_{v,i+1}, u_{v,i}) | i = 1, \ldots, k-1\}. \end{aligned} \tag{7}$$

The capacity of each constraint edge is set to infinity:

$$c(u_{v,i+1}, u_{v,i}) = +\infty, \ i = 1, \ldots, k-1.$$

This precludes more than one data edge being in the cut in each column $\mathcal{E}_{\mathrm{D}}^{v}$ of data edges over vertex v. To see this, consider how the assignments of vertices to S or T changes as we proceed from s to t on a column, remembering a cut is simply an assignment of each vertex to one of S or T. The first vertex s belongs to S, and the last vertex t belongs to T. Thus, there must be at least one boundary in the progression where the membership changes. Moreover, the direction of the change must alternate (if you go from S to T, next time you have to come back from T to S.) Suppose that there is more than one boundary. Then the change across at least one of them must be from T to S. There is an edge going each way at this boundary. The edge from S to T is a constraint edge, and by the definition of a cut, the constraint edge is in the cut. Therefore, if constraint edges have infinite capacities, there cannot be more than one boundary on the column. In Figure 1, constraint edges are depicted as dashed arrows, and none is in the cut.

3.5 Interpretation of Cuts

As discussed in the previous subsection, the constraint edges guarantee that in each column there is exactly one edge in the cut. Because of this, we can interpret a cut as a configuration of the MRF. Remember that the space $\mathcal{X} = L^V$ is a set of configurations $X : V \ni v \mapsto X_v \in L$, and we want to find the configuration X that minimizes the energy (3).

> **Convention.** Given any cut of $\mathcal{G}$ with finite total cost, for each $v \in G$ let $u_{v,i}$ be the vertex at the tail of the uniquely determined edge in the cut for the column over v. We interpret the cut as a configuration X such that $X_v = i$.

If an MRF configuration $X \in \mathcal{X}$ assigns label i to vertex v, the data edge $(u_{v,i}, u_{v,i+1})$ (or if $i = k$, $(u_{v,i}, t)$), whose cost is $h(v, l_i)$ by (6) is in the corresponding cut. Thus, with the capacities of the edges defined so far, the total cost of any cut is, if finite, exactly the second sum in the energy (3) for the MRF configuration.

The rest of the energy, the prior term, is realized by the cost of the cut of a third kind of edge, the penalty edges.

3.6 Penalty Edges

Penalty edges go between columns:

$$\mathcal{E}_{\mathrm{P}} = \{(u_{v,i}, u_{w,j}) | \, (v, w) \in E; \; i, j \in L\}.$$

Figure 1 shows the special case where the capacity of such an edge $(u_{v,i}, u_{w,j})$ is zero unless $i = j$; that is, where only horizontal penalty edges exist. It is easily seen that the number of horizontal edges that are in the cut between columns over neighboring vertices v and w is proportional to the change $|X_v - X_w|$ of assignments at the vertices. For instance, if the assignment does not change, none of the horizontal edges between the two columns is in the cut. If they change by one, one would be in the cut, as at the leftmost pair of columns in Figure 1. This is the simplest case. Now, we consider the general case where each pair $(u_{v,i}, u_{w,j})$ of vertices in neighboring columns can have an edge between them with a nonnegative capacity (see Figure 2.)

It is important to note that specifying an assignment at a vertex v, which under the convention determines the unique data edge that is in the cut in the column over v, also determines the membership in either S or T of all vertices in the column. That is, if $X_v = i$, the vertices above $u_{v,i}$ belong to T, and the rest of the column belong to S. Therefore, given assignments X_v and X_w at neighboring vertices v and w of the original graph G, the status (in the cut or not) of each edge between vertices in the columns over these vertices is determined. Thus, the sum of the capacities of the edges in the cut between the two columns is determined by X_v and X_w.

Suppose we have label $X_v = i$ at vertex v and $X_w = j$ at vertex w. By the convention, this is expressed as a cut that has $(u_{v,i}, u_{v,i+1})$ and $(u_{w,j}, u_{w,j+1})$ in it. As a side effect, various other edges are also in the cut. Among the edges going from the column over v to the one over w, edges of the form $(u_{v,a}, u_{w,b})$ with $a \leq i$ and $b > j$ are in the cut. Similarly, edges of the form $(u_{w,b}, u_{v,a})$ with $a > i$ and $b \leq j$ are in the cut. The sum of the capacities of these

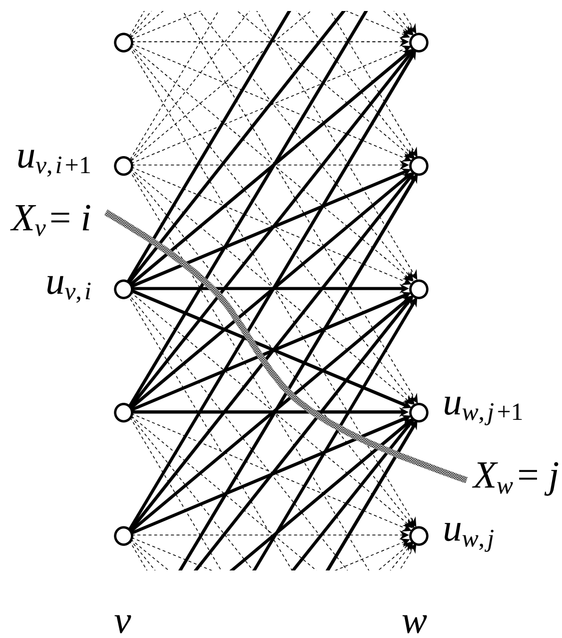

Figure 2: General penalty edges. Only the edges originating from the column over vertex v are shown. Edges going from "below" $u_{v,i+1}$ to "above" $u_{w,j}$ are in the cut, shown here as solid arrows.

edges amounts to:

$$f(i,j) = \sum_{a=1}^{i} \sum_{b=j+1}^{k} c(u_{v,a}, u_{w,b}) + \sum_{a=i+1}^{k} \sum_{b=1}^{j} c(u_{w,b}, u_{v,a}). \tag{8}$$

Now, we *assume* that the sum $f(i,j)$ depends only on the difference of the labels $i - j$. Under this assumption we define a function $\tilde{g}(x)$ by

$$\tilde{g}(i-j) = f(i,j).$$

Although we do not even know if there exists a capacity function c that makes the sum (8) depend only on the difference between i and j (we will see below that it does exist,) here we just assume such capacity does exist and derive a necessary condition.

Proposition. *Assume that the function $f(i,j)$, defined by (8) as the sum of edge capacities between neighboring columns, depends only on the difference between i and j, and thus can be written as $f(i,j) = \tilde{g}(i-j)$ with a function $\tilde{g}(x)$. Then $\tilde{g}(x)$ is convex.*

Proof. The second difference of $\tilde{g}$ is

$$\{\tilde{g}(x+1)-\tilde{g}(x)\}-\{\tilde{g}(x)-\tilde{g}(x-1)\}=f(i,j-1)-f(i,j)-f(i-1,j-1)+f(i-1,j),$$

where $i-j=x$. When we use the definition (8) and simplify the sum, the second difference is

$$c(u_{v,i},u_{w,j})+c(u_{w,j},u_{v,i}). \tag{9}$$

Since the edge capacity is nonnegative, this shows that the function $\tilde{g}(x)$ must have a nonnegative second difference; thus $\tilde{g}(x)$ is convex. □

Conversely, suppose we are given a convex function $g(x)$. (In the rest of this section, we assume $\alpha_{uv}=1$, since we only look at a particular pair of vertices, and the necessary modification is obvious even when α_{uv} depends on (u,v).) We will show that this function can be realized as the sum (8) of the penalty edges with suitably defined capacities. But we almost know the capacities: if it can be done at all, $g(x)$ has to have (9) as its second difference. The trick is to make the sum $f(i,j)$ in (8) depend only on the difference $i-j$.

Note the following two points. First, since L is finite, the range of the target function (i.e., any assignment of labels to vertices) is bounded, and so is the range of possible x, which is the difference between the two values of the target function at neighboring vertices. In other words, the behavior of $g(x)$ outside of this range does not make any difference to the problem. Specifically, we can assume that the second derivative of g is zero for large enough $|x|$. Second, since $g(X_u-X_v)$ and $g(X_v-X_u)$ always appear together in the energy (3), we can assume that $g(x)$ is symmetric; otherwise, we can replace $g(x)$ with $(g(x)+g(-x))/2$ without changing the energy.

Now, we define the capacities of the edges by half the second difference of $g(x)$:

$$c(u_{v,i},u_{w,j})=\frac{g(i-j+1)-2g(i-j)+g(i-j-1)}{2}, \tag{10}$$

where the right hand side is nonnegative because $g(x)$ is convex. Thus, the capacity of an edge depends only on the label change $i-j$ between the head and the tail of the edge. Therefore, when there is no boundary effect, i.e., when the columns are of infinite height, the sum also

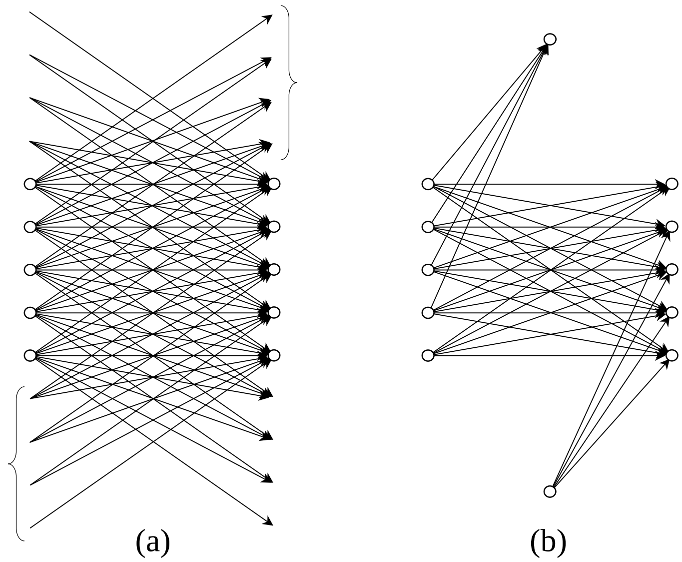

Figure 3: (a) Each penalty edge has a capacity that depends only on the label change. (b) Contributions from out-of-bounds edges are consolidated.

would only depend on the difference $i - j$. We know that the sum is finite because of the assumption that the second derivative of $g(x)$ vanishes for sufficiently large $|x|$ and we can thus ignore edges with a large label difference.

The columns, however, have top and bottom. That is, for this argument to work, we have to add the capacities of edges that have an end that is out of bounds (see Figure 3 (a).) The problem is that some edges whose capacities are necessary to make the sum depend only on the label change do not actually exist. Note however that there are only a finite number of such edges because the capacity is zero for sufficiently large label differences. Therefore we can just add these capacities to those of other edges that do exist (Figure 3 (b)) without changing the total cost of any cut. That is, whenever vertices of the form $u_{v,i}$ with $i > k$ appear, we replace them by t and add the capacity to the existing edge; similarly, we replace $u_{v,i}$ by $u_{v,1}$ if $i < 1$. Note that this does not mean that the result is approximate; the total cost of any cut stays the same.

Thus we can make the sum $f(i,j)$ in (8) depend only on the difference $i - j$. This realizes a function $\tilde{g}(x)$ by $\tilde{g}(i-j) = f(i,j)$. Following the reasoning in the proof of the Proposition, we obtain the second difference of $\tilde{g}(x)$ as in (9). Substituting the capacity (10), and noting that $g(x)$ is convex, we see that the second differences of $\tilde{g}(x)$ and $g(x)$ coincide.

In this way, for a given convex function $g(x)$, we can realize the sum (8) that has exactly the same second difference. In other words, $g(x)$ is realized by the sum up to a difference of a constant and a linear term in x. Since $g(X_u - X_v)$ and $g(X_v - X_u)$ always appear together in the energy (3), the linear difference cancels out. The constant difference results in a constant in the energy, since there are a fixed number of vertices in the sum, and is therefore immaterial to the optimization. Thus, by finding a minimum cut, we can find a global optimum of the MRF. We have proved:

Theorem. *Given a label set* $L = \{l_1, \ldots, l_k\}$ *and a first order MRF defined by a graph* $G = (V, E)$ *and the energy functional*

$$E(X) = \sum_{(u,v)\in E} \alpha_{uv}\, g(\iota(X_u) - \iota(X_v)) + \sum_{v\in V} h(v, X_v)$$

on the space of assignments $V \ni v \mapsto X_v \in L$ *with an index function* $\iota : l_i \mapsto i$*, a convex function* $g(x)$*, and an arbitrary function* $h(v,l)$*, define a graph* $\mathcal{G}$ *as above. Then, cuts of* $\mathcal{G}$ *with finite costs and configurations of the MRF are in a one-to-one correspondence that preserves the order of respective cost and energy; and a minimum cut therefore corresponds to a global minimum energy configuration of the MRF. Moreover, the convexity of* $g(x)$ *is necessary and sufficient for the nonnegativeness of all edge capacities in* $\mathcal{G}$*, which is sufficient for computation of minimum cuts in polynomial time.*

4 Conclusion

In this paper, we have introduced a method that maps certain MRF optimization problems into graph problems that have efficient solutions. Because of the use of the minimum cut algorithm, a globally optimal solution is found in polynomial time. The method can solve exactly first order Markov Random Field problems in more generality than was previously possible. We also show that the convexity of prior function in the energy is both necessary and sufficient for the applicability of the method.

The principal limitation of the method is the convexity criterion on the prior function. Another limitation is the condition that the label set should be such that placing the labels in a linear order makes sense in terms of the prior. This is a problem for instance when restoring a color image. Otherwise, the method is completely general.

Acknowledgements

The author wishes to thank Professor Davi Geiger for his advice and encouragement. This work was supported in part by NSF Grant No. IIS-9733913.

References

[1] A. A. Amini, T. E. Weymouth, and R. C. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **12**(9):855–867, 1990.

[2] H. H. Baker and T. O. Binford. Depth from Edge and Intensity-based Stereo. In *Proceedings of 7th International Joint Conferences on Artificial Intelligence*, 631–636, 1981.

[3] J. Besag. Spatial interaction and the statistical analysis of lattice systems (with discussion). *Journal of the Royal Statistical Society, Series B*, **36**:192–326, 1974.

[4] Y. Boykov, O. Veksler, and R. Zabih. Efficient Approximate Energy Minimization via Graph Cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-20(12):1222–1239, November 2001.

[5] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to algorithms*. McGrow-Hill. New York. 1990.

[6] D. Geiger, A. Gupta, L. Costa, and J. Vlontzos. Dynamic Programming for Detecting, Tracking, and Matching Deformable Contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(3):294–302, 1993.

[7] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**:721–741, 1984.

[8] D. M. Greig, B. T. Porteous, and A. H. Seheult. Discussion of: On the statistical analysis of dirty pictures (by J. E. Besag.). *Journal of the Royal Statistical Society, Series B*, **48**:282–284, 1986.

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

[10] H. Ishikawa. Multi-scale Feature Selection in Stereo. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO, U.S.A. 132–137, 1999.

[11] H. Ishikawa. *Global Optimization Using Embedded Graphs.* Doctoral dissertation, Department of Computer Science, New York University, May 2000.

[12] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *Fifth European Conference on Computer Vision*, Freiburg, Germany. 232–248, Springer-Verlag. 1998.

[13] H. Ishikawa and D. Geiger. Segmentation by Grouping Junctions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA, U.S.A. 125–131, 1998.

[14] H. Ishikawa and D. Geiger. Mapping Image Restoration to a Graph Problem. In *IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing*, Antalya, Turkey. 1999.

[15] R. Kinderman and J. L. Snell. *Markov Random Fields and Their Applications*. American Mathematical Society. Providence, RI, U.S.A. 1980.

[16] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, **21**:1087–1091, 1953.

[17] U. Montanari. On the Optimal Detection of Curves in Noisy Pictures. *Communications of the ACM*, **15**(5):335–345, 1971.

[18] S. Roy. Stereo without epipolar lines : A maximum-flow formulation. *International Journal of Computer Vision*, 34:147–162, 1999.

[19] S. Roy and I. Cox. A maximum-flow formulation of the N-camera stereo correspondence problem. In *Proceedings of the International Conference on Computer Vision*, Bombai, India.492–499, 1998.