Improved Moves for Truncated Convex Models

M. Pawan Kumar   PAWAN@CS.STANFORD.EDU
Computer Science Department
Stanford University
Stanford, CA 94305, USA

Olga Veksler   OLGA@CSD.UWO.CA
Computer Science Department
University of Western Ontario
London, ON N6A 5B7, Canada

Philip H.S. Torr   PHILIPTORR@BROOKES.AC.UK
Department of Computing
Oxford Brookes University
Oxford, OX33 1HX, UK

Abstract

We consider the problem of obtaining an approximate maximum *a posteriori* estimate of a discrete random field characterized by pairwise potentials that form a truncated convex model. For this problem, we propose two *st*-MINCUT based move making algorithms that we call Range Swap and Range Expansion. Our algorithms can be thought of as extensions of αβ-Swap and α-Expansion respectively that fully exploit the form of the pairwise potentials. Specifically, instead of dealing with one or two labels at each iteration, our methods explore a large search space by considering a range of labels (that is, an interval of consecutive labels). Furthermore, we show that Range Expansion provides the same multiplicative bounds as the standard linear programming (LP) relaxation in polynomial time. Compared to previous approaches based on the LP relaxation, for example interior-point algorithms or tree-reweighted message passing (TRW), our methods are faster as they use only the efficient *st*-MINCUT algorithm in their design. We demonstrate the usefulness of the proposed approaches on both synthetic and standard real data problems.

Keywords: truncated convex models, move making algorithms, range moves, multiplicative bounds, linear programming relaxation

1. Introduction

Discrete pairwise random fields are a useful tool for concisely specifying the probability of a labeling (that is, an assignment of values) for a set of discrete random variables. Hence, they offer an elegant formulation for several problems in computer vision, from low-level tasks such as stereo reconstruction and image denoising (Szeliski et al., 2008) to high-level tasks such as pose estimation (Felzenszwalb and Huttenlocher, 2000) and scene segmentation (Shotton et al., 2006). Once formulated within this framework, the problem is typically solved by obtaining the maximum *a posteriori* (MAP) estimate, that is, finding the labeling that minimizes the corresponding Gibbs energy (hereby referred to as simply the energy). This is well-known to be an NP-hard problem and thus, requires us to come up with accurate approximation algorithms.
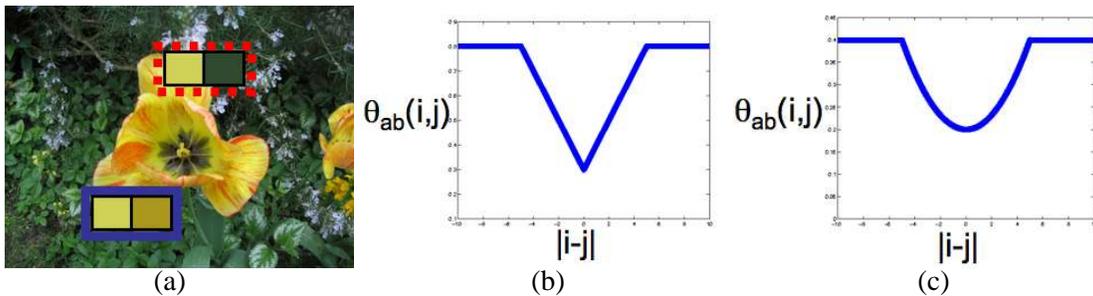
Figure 1: (a) An example of a natural image that consists of smoothly varying intensities (for instance, the two enlarged pixels bounded in the solid box near the bottom of the image; shown in blue if viewed in color) and sharp edges (for instance the two enlarged pixels bounded in the dashed box near the top of the image; shown in red if viewed in color). The smooth variation is captured by the convex part of truncated convex models. The sharp edges are not over penalized due to the truncation, thereby making the potentials robust. (b)-(c) Two examples of truncated convex potentials that will be of interest to us in this work: truncated linear metric (b) and truncated quadratic semi-metric (c).

It is common practice in computer vision to specify an energy function with arbitrary unary potentials and truncated convex pairwise potentials (Boykov et al., 2001; Szeliski et al., 2008; Veksler, 1999). This is especially true in low-level vision where the use of truncated convex models is motivated by the fact that pixels belonging to the same *object* are similar in appearance—captured by the convex part of the pairwise potentials—while pixels belonging to different objects induce an edge in the image—captured by the truncated part (see Fig. 1). In other words, convexity encourages smoothness while truncation ensures that edges are not over penalized. Given their widespread use, the problem of MAP estimation for truncated convex models merits special attention.

In this work, we develop two approaches, called Range Swap and Range Expansion, that take advantage of the special form of the pairwise potentials to obtain an accurate MAP estimate. Specifically, our methods iteratively minimize the energy by searching over a subset of the possible labelings specified by the original problem. Each iteration is formulated as an *st*-MINCUT problem for which there exist several efficient algorithms (Boykov and Kolmogorov, 2004). Unlike other *st*-MINCUT based approaches (Boykov et al., 2001) that restrict the number of labels for each random variable at an iteration to at most 2, our methods explore a large search space by considering a range of labels (that is, an interval of consecutive labels). Our methods are both practically useful and theoretically interesting: in practice, they provide an improved performance (lower energy labelings); in theory, we show that Range Expansion provides the same guarantees as the standard linear programming (LP) relaxation in polynomial time. Specifically, it obtains the LP relaxation's multiplicative bounds for the truncated linear and truncated quadratic pairwise potentials. Note that this does not imply that it provides the same solution as the LP relaxation. However, as our experiments will demonstrate, they provide comparable results (the LP relaxation typically provides lower energy values but at a high computational cost).

Before proceeding further, we would like to note here that the algorithms presented in this paper can be trivially extended to *truncated submodular models*, where submodularity is as defined in Schlesinger and Flach (2006) and is a strict generalization of convexity (Ishikawa, 2003). However, we will restrict our discussion to truncated convex models as it makes the description and analysis of our methods simpler. For clarity of presentation, many of the proofs are reported in the Appendix.

Preliminary versions of this paper have appeared as Kumar and Torr (2008) and Veksler (2007). The project webpage is located at the following URL:

http://ai.stanford.edu/~pawan/research/truncated-moves.html.

2. Related Work

Given the popularity of truncated convex models, it is not surprising that the corresponding MAP estimation problem has been well-studied in the literature. For example, Felzenszwalb and Huttenlocher (2004) improved the efficiency of the popular max-product belief propagation (BP) algorithm Pearl, 1988 by using the special form of the pairwise potentials. Note that BP provides the exact MAP estimate for tree-structured random fields. However, for a general neighborhood structure, BP is not guaranteed to converge.

The results of Felzenszwalb and Huttenlocher (2004) can be used directly to speed-up the tree-reweighted message passing algorithm (TRW) (Wainwright et al., 2005) and its sequential variant TRW-S (Kolmogorov, 2006). Both TRW and TRW-S attempt to optimize the Lagrangian dual of the standard LP relaxation of the MAP estimation problem (Chekuri et al., 2005; Koster et al., 1998; Schlesinger, 1976; Wainwright et al., 2005). Unlike BP and TRW, TRW-S is guaranteed to converge. However, TRW-S and other related algorithms (Globerson and Jaakkola, 2007; Komodakis et al., 2007; Schlesinger and Giginyak, 2007a,b) suffer from the following problems: (i) An extensive comparison of energy minimization algorithms by Szeliski et al. (2008) revealed that TRW-S is slower than *st*-MINCUT based algorithms. Other approaches, such as dual coordinate ascent (Globerson and Jaakkola, 2007) or dual decomposition (Komodakis et al., 2007), are even slower than TRW-S in practice (even though, unlike TRW-S, dual decomposition is capable of escaping from the *weak tree agreement* local minimum). (ii) TRW-S and the related methods attempt to solve the dual of the LP relaxation. When the dual is not *decodable* (that is, when the LP relaxation is not tight for a specific instance of the problem), the primal solution is often obtained in a heuristic fashion (for example, by using monotonic chain decoding Meltzer et al., 2005).

Another way of solving the LP relaxation is to resort to interior point algorithms (Boyd and Vandenberghe, 2004) or iterative Bregman projections (Ravikumar et al., 2008). These approaches provide the primal (possibly fractional) solution of the LP relaxation, but at a high computational cost. In our experience, the existing software for interior point algorithms is unable to deal with energy minimization problems for moderately sized (620×480) images. However, if a primal solution can be obtained then certain randomized rounding schemes provide the following guarantees (Chekuri et al., 2005):

- For Potts model, a multiplicative bound of 2 is obtained by using the rounding scheme of Kleinberg and Tardos (1999).

- For the truncated linear metric, a multiplicative bound of $2 + \sqrt{2}$ is obtained using the rounding scheme of Chekuri et al. (2005).

- For the truncated quadratic semi-metric, a multiplicative bound of $O(\sqrt{M})$ is obtained using the rounding scheme of Chekuri et al. (2005). Here M is the truncation factor.

The algorithms most related to our approach are the so-called move making methods that rely on solving a series of *st*-MINCUT problems. Move making algorithms start with an initial labeling and iteratively minimize the energy by moving to a better labeling. At each iteration, (a subset of) random variables have the option of either retaining their old label or taking a new label from a subset of the labels **l**. For example, in the αβ-swap algorithm (Boykov et al., 2001) the variables currently labeled l_α or l_β can either retain their labels or swap them (that is, some variables labeled l_α can be relabeled as l_β and vice versa). In the α-expansion algorithm (Boykov et al., 2001), each variable can either retain its label or get assigned the label l_α during an iteration. Unlike αβ-swap, which has no guarantees on the quality of its solution, the α-expansion algorithm and its generalization using a primal-dual scheme (Komodakis and Tziritas, 2007) provide the following bounds:

- For the Potts model, a multiplicative bound of 2 is obtained using α-expansion (Boykov et al., 2001).

- For the truncated linear metric, a multiplicative bound of $2M$ is obtained using α-expansion (Boykov et al., 2001).

- For the truncated quadratic semi-metric, a multiplicative bound of $2M$ is obtained using the primal-dual scheme of Komodakis and Tziritas (2007).

It is also worth noting that we can obtain a bound of 2 for the related multiway cut problem (Vazirani, 2001) using the *st*-MINCUT algorithm.

Both αβ-swap and α-expansion only allow a variable to take one of two possible labels at each iteration. In other words, they are restricted to a small search space during each move. Gupta and Tardos (2000) extended the α-expansion algorithm for the truncated linear metric by considering a range of labels and provided a multiplicative bound of 4. However, their method is not applicable for the case of truncated quadratic semi-metric. Note that the bounds obtained by all the above move making algorithms are inferior to the bounds obtained by the LP relaxation for truncated convex models (as summarized in table 1). In fact, a recent result shows that the bounds obtained by Boykov et al. (2001) and Komodakis and Tziritas (2007) can also be achieved using the simple iterated conditional modes (ICM) algorithm Gould et al., 2009. However, despite providing inferior bounds, move making algorithms use only a single *st*-MINCUT at each iteration and hence, are often faster than interior point algorithms, TRW, TRW-S and BP.

3. Preliminaries

Before providing the details for Range Swap and Range Expansion, we set up the notation and briefly review some preliminary concepts that are used in the remainder of the paper.

3.1 Random Field

Given data **D** (for example, an image or a video), random fields model the probability of a set of random variables **v**, that is, either the joint distribution of **v** and **D** as in the case of Markov random fields (MRF) (Besag, 1986) or the conditional distribution of **v** given **D** as in the case of conditional

	α-exp	PD	GT	LP	Our (Range Expansion)
Potts	2	2	-	2	-
Trunc. Linear	2M	2M	4	$2 + \sqrt{2}$	$2 + \sqrt{2}$
Trunc. Quad.	-	2M	-	$O(\sqrt{M})$	$O(\sqrt{M})$

Table 1: The multiplicative bounds obtained by various algorithms for the three commonly used truncated convex models. PD refers to the primal-dual method of Komodakis and Tziritas (2007), GT refers to the method by Gupta and Tardos (2000) and LP refers to the multiplicative bounds obtained by the LP relaxation. Note that, unlike our approach, previous move making algorithms provide inferior bounds compared to LP for truncated linear metric and truncated quadratic semi-metric.

random fields (CRF) (Lafferty et al., 2001). The word 'discrete' refers to the fact that each of the random variables $v_a \in \mathbf{v} = \{v_0, \cdots, v_{n-1}\}$ can take one label from a discrete set $\mathbf{l} = \{l_0, \cdots, l_{h-1}\}$. Throughout this paper, we will assume an MRF framework while noting that our results are equally applicable for a CRF.

An MRF defines a neighborhood relationship (denoted by $\mathcal{E}$) over the random variables such that $(a, b) \in \mathcal{E}$ if v_a and v_b are neighboring random variables. Given an MRF, a labeling refers to a function f such that

$$f : \{0, \cdots, n-1\} \longrightarrow \{0, \cdots, h-1\}.$$

In other words, the function f assigns the label $l_{f(a)} \in \mathbf{l}$ to each random variable $v_a \in \mathbf{v}$. The probability of the labeling is given by the following Gibbs distribution:

$$\Pr(f, \mathbf{D}|\theta) = \frac{1}{Z(\theta)} \exp(-Q(f, \mathbf{D}; \theta)), \tag{1}$$

where θ is the parameter vector of the MRF and $Z(\theta)$ is the partition function. Since we consider pairwise MRFs, the energy can be written as:

$$Q(f, \mathbf{D}; \theta) = \sum_{v_a \in \mathbf{v}} \theta_a(f(a)) + \sum_{(a,b) \in \mathcal{E}} \theta_{ab}(f(a), f(b)).$$

Here, $\theta_a(f(a))$ denotes unary potentials and $\theta_{ab}(f(a), f(b))$ denotes pairwise potentials, that is, $\theta_a(f(a))$ is the cost of assigning label $l_{f(a)}$ to variable v_a and $\theta_{ab}(f(a), f(b))$ is the cost of assigning labels $l_{f(a)}$ and $l_{f(b)}$ to variables v_a and v_b respectively. Using Equation (1) it follows that the labeling f that maximizes the posterior $\Pr(f, \mathbf{D}; \theta)$ (that is, the MAP estimate) can be obtained by minimizing the energy.

3.2 Truncated Convex Models

We consider the problem of MAP estimation of random fields where the pairwise potentials are defined by truncated convex models (Veksler, 1999). Formally speaking, the pairwise potentials are of the form

$$\theta_{ab}(f(a), f(b)) = w_{ab} \min\{d(f(a) - f(b)), M\},$$

where $w_{ab} \geq 0$ for all $(a,b) \in \mathcal{E}$, $d(\cdot)$ is a convex function and $M > 0$ is the truncation factor. Here, the term 'convex' is used according to the definition of Ishikawa (2003). Specifically, a function $d(\cdot)$ defined over integers is convex if, and only if,

$$d(x+1) - 2d(x) + d(x-1) \geq 0, \forall x \in \mathbb{Z}.$$

It is assumed that $d(x) = d(-x)$. Examples of pairwise potentials of this form include the truncated linear metric and the truncated quadratic semi-metric, that is,

$$\theta_{ab}(f(a), f(b)) = w_{ab} \min\{|f(a) - f(b)|, M\},$$
$$\theta_{ab}(f(a), f(b)) = w_{ab} \min\{(f(a) - f(b))^2, M\}.$$

An illustration of the above potentials is provided in Fig. 1(b)-(c).

3.3 Multiplicative Bounds

The worst case accuracy of a MAP estimation approach can be expressed using its multiplicative bound. Formally, let f be the labeling obtained by an algorithm A (randomized or deterministic) for an instance of the MAP estimation problem belonging to a particular class (in our case when the pairwise potentials form a truncated convex model). Let f^* be the optimal labeling. The algorithm A is said to achieve a multiplicative bound of σ if for every instance in the specific class the following holds true:

$$E\left(\frac{Q(f, \mathbf{D}; \theta)}{Q(f^*, \mathbf{D}; \theta)}\right) \leq \sigma,$$

where $E(\cdot)$ denotes the expectation and can be dropped from the above inequality if the algorithm is deterministic (as in our case).

3.4 The st-MINCUT Problem

Given a directed, non-negatively weighted graph with two terminal vertices s (the source) and t (the sink), an st-cut is defined as a partitioning of the vertices of the graph into two disjoint sets such that the first partition contains s while the second partition contains t. The st-MINCUT problem is to find the minimum cost st-cut, where the cost of a cut is measured as the sum of the weights of the edges whose starting point belongs to the first partition and ending point belongs to the second partition. It is well-known that the st-MINCUT problem can be formulated as a linear program (LP) (which is different but closely related to the LP relaxation for the general MAP estimation problem) with integer solutions. The st-MINCUT problem has several efficient polynomial and pseudo-polynomial solvers (Boykov and Kolmogorov, 2004; Dinic, 1970; Goldberg and Tarjan, 1988). In this work, we employ the pseudo-polynomial solver of Boykov and Kolmogorov (2004) that has been shown to have a linear complexity in practice for several computer vision tasks. The low complexity of this algorithm is responsible for making our iterative algorithm (which solves an st-MINCUT problem at each iteration) computationally efficient.

In order to help the reader follow the arguments of the paper, we provide the list of terms used throughout the paper along with their meanings in table 2.

$\mathbf{D}$	Data provided by the user (for example, an image or a video).
n	Number of random variables.
$\mathbf{v}$	Set of random variables $\{v_0, \cdots, v_{n-1}\}$.
$\mathcal{E}$	Set of neighboring random variables v_a and v_b (denoted by $(a,b) \in \mathcal{E}$).
h	Number of labels.
$\mathbf{l}$	Set of labels $\{l_0, \cdots, l_{h-1}\}$.
I_m	Interval of consecutive labels $[i_m + 1, j_m]$.
L	Length of the interval, that is, $L = j_m - i_m$.
I_r	Set of intervals $\{[0,r],[r+1,r+L],\cdots,[.,h-1]\}$.
f	Labeling of the random field (v_a takes a label $l_{f(a)}$).
f^*	An optimal (MAP) labeling of the random field.
$\theta_a(i)$	Unary potential of assigning label l_i to v_a.
w_{ab}	Weight for neighboring random variables $(a,b) \in \mathcal{E}$.
$d(\cdot)$	Convex function used to define the distance between two labels.
$\hat{d}(\cdot)$	$\hat{d}(x) = d(x+1) - d(x) - d(1) + d(0)/2$.
M	Truncation factor.
κ_{ab}	Constant $w_{ab}d(L)$.
$\theta_{ab}(i,j)$	The pairwise potential $w_{ab}\min\{d(i-j),M\}$ of assigning labels l_i and l_j to neighboring random variables v_a and v_b respectively.
θ	Parameter vector of the discrete random field.
$Q(f;\mathbf{D},\theta)$	Energy of the labeling f given the data $\mathbf{D}$ and parameters θ.
S	Index for a subset of random variables $S \subseteq \{0,1,\cdots,n_1\}$.
$\mathbf{v}(S)$	$\{v_a \in \mathbf{v}, a \in S\}$.
$\mathcal{A}(S)$	$\{(a,b) \in \mathcal{E}, a \in S, b \in S\}$.
$\mathcal{B}_1(S)$	$\{(a,b) \in \mathcal{E}, a \in S, b \notin S\}$.
$\mathcal{B}_1(S)$	$\{(a,b) \in \mathcal{E}, a \notin S, b \in S\}$.
$\mathcal{B}(S)$	$\mathcal{B}_1(S) \cup \mathcal{B}_2(S)$.
$\mathbf{v}(f,I_m)$	$\{v_a \in \mathbf{v}, f(a) \in I_m\}$.
$\mathcal{A}(f,I_m)$	$\{(a,b) \in \mathcal{E}, f(a) \in I_m, f(b) \in I_m\}$.
$\mathcal{B}_1(f,I_m)$	$\{(a,b) \in \mathcal{E}, f(a) \in I_m, f(b) \notin I_m\}$.
$\mathcal{B}_2(f,I_m)$	$\{(a,b) \in \mathcal{E}, f(a) \notin I_m, f(b) \in I_m\}$.
$\mathcal{B}(f,I_m)$	$\mathcal{B}_1(f,I_m) \cup \mathcal{B}_2(f,I_m)$.
$\mathcal{G}_m$	Graph corresponding to I_m over which an *st*-MINCUT problem is defined.
$\mathcal{V}_m$	Set of vertices a_k and $b_{k'}$ for $\mathcal{G}_m$ such that $k,k' \in I_m$.
$\mathcal{E}_m$	Set of edges $(a_k,b_{k'})$ for $\mathcal{G}_m$.

Table 2: List of the various terms used throughout the paper.

4. Why Range Moves?

As mentioned earlier, our methods differ from previous move making approaches that deal with only 1 or 2 (not necessarily consecutive) labels at each iteration by considering a range of labels. In other words, we obtain a local minimum labeling with respect to a large search space defined by intervals of consecutive labels. To motivate our choice of using a range of labels, we show that an algorithm that obtains the local minimum with respect to *smooth labelings* provides a small multiplicative bound and hence, a tight approximation. Before proceeding further, we require the following definitions.

Definition 1: Let $S \subseteq \{0, \cdots, n-1\}$ be a subset of the indices of the random variables. A labeling f is said to be *smooth* with respect to S if, and only if, for each $(a,b) \in \mathcal{E}$ such that $a \in S$ and $b \in S$, there exists a path $a_0 = a, a_1, \cdots, a_q = b$ such that $(a_i, a_{i+1}) \in \mathcal{E}$, $a_i \in S$ and $d(f(a_i) - f(a_{i+1})) \leq M$ for all $i = 0, 1, \cdots, q-1$. In other words, the pairwise potential for each edge in the path lies in the convex part (indicating the lack of a discontinuity, hence the name smooth labeling). Note that this does not necessarily imply that $d(f(a), f(b)) \leq M$.

Definition 2: A labeling $\hat{f}$ is said to be a local minimum over smooth labelings if the energy cannot be reduced further by changing the labels of any subset of random variables, say defined by S, such that the new labeling f is smooth with respect to S. In other words, if $f(a) = \hat{f}(a)$ for all $a \notin S$ and f is smooth with respect to S, then $Q(\hat{f}, \mathbf{D}; \theta) \leq Q(f, \mathbf{D}; \theta)$, for all $S \subseteq \{0, \cdots, n-1\}$.

Using the above definitions, we can state the following theorem.

Theorem 1: An algorithm that provides a local minimum over smooth labelings achieves a multiplicative bound of 2 (Proof in Appendix A).

Note that a multiplicative bound of 2 is superior to the best known approximation guarantees (obtained by the LP relaxation). However, an algorithm that provides the desired local minimum labeling would be computationally infeasible. To see why, consider a random field with three variables v_a, v_b and v_c that are neighbors of each other. Suppose there exists a labeling f such that $d(f(a) - f(b)) \leq M$, $d(f(b) - f(c)) \leq M$ and $d(f(a) - f(c)) > M$. Note that this labeling is smooth since we can find a path from v_a to v_c via v_b such that the edges in the path lie in the convex part. In order to obtain a local minimum over smooth labelings, an algorithm needs to be able to search over such labelings f (that is, provide the optimal move over all smooth labelings). This implies that the algorithm should be able to solve the problem of MAP estimation in the presence of truncation (since $\theta_{ac}(f(a), f(c))$ would lie in the truncated part). Since MAP estimation in truncated convex models is an NP-hard problem, such an algorithm would not be computationally feasible unless P = NP.

Although the above argument shows that we will not be able to design an algorithm that provides a local minimum over smooth labelings, it serves to demonstrate the benefit of allowing each random variable to choose from a range of labels. Even though the range cannot be large enough to cover all smooth labelings, we should at least explore as large a subset of labelings as is computationally feasible. Clearly, this is an issue that is not considered in previous move making approaches. In order to alleviate this deficiency, we develop two algorithms that consider a large range of labels for each random variable. Table 3 describes the main steps involved in both the algorithms. The two methods differ in the way they move from one labeling to the next. In the next two sections, we provide a detailed description and analysis of our methods.

Initialization
- Initialize the labeling to some function f_1. For example, $f_1(a) = 0$ for all $v_a \in \mathbf{v}$.

Iteration
- Set $i_m = 0$ (where i_m indexes the interval to be used).
- While $i_m < h$
  - Define interval $I_m = [i_m + 1, j_m]$ where $j_m = \min\{i_m + L, h - 1\}$ and $d(L) \geq M$.
  - Move from current labeling f_m to a new labeling f_{m+1} using st-MINCUT such that
    (i) if $f_{m+1}(a) \neq f_m(a)$ then $f_{m+1}(a) \in I_m$,
    (ii) $Q(f_{m+1}, \mathbf{D}; \theta) \leq Q(f_m, \mathbf{D}; \theta)$.
  - $i_m \leftarrow i_m + 1$.

Termination
- Stop if energy does not decrease for any interval I_m, otherwise repeat **Iteration**.

Table 3: As is typical with move making methods, our methods iteratively move from one labeling to the next by solving an st-MINCUT problem. They are said to converge when there remain no moves that reduce the energy further. The two algorithms, Range Swap and Range Expansion, differ in the way they choose the new labeling f_{m+1}. Specifically, they construct different graphs for the corresponding st-MINCUT problem.

5. The Range Swap Algorithm

Range Swap can be thought of as an appropriate modification of the $\alpha\beta$-swap algorithm of Boykov et al. (2001) for truncated convex models. At an iteration m, the Range Swap algorithm only considers the random variables v_a whose current labeling $f_m(a)$ lies in the interval $I_m = [i_m + 1, j_m]$ of length L.[1] In order to simplify the explanation of the algorithm, we will begin by assuming that $d(L) = M$ and later relax this condition such that $d(L) \geq M$. Keeping the labels of all other random variables fixed, Range Swap provides the option for random variables with $f_m(a) \in I_m$ to change their labels to $f_{m+1}(a) \in I_m$ (or retain their current label). In order to provide a concrete description of the algorithm, we define a set $S_m = \{a | f_m(a) \in I_m\}$. Using S_m we define the set of random variables $\mathbf{v}(S_m)$ and the set of edges $\mathcal{A}(S_m)$, $\mathcal{B}_1(S_m)$, $\mathcal{B}_2(S_m)$ and $\mathcal{B}(S_m)$ as follows:

$$
\begin{aligned}
\mathbf{v}(S_m) &= \{v_a | a \in S_m\}, \\
\mathcal{A}(S_m) &= \{(a,b) | (a,b) \in \mathcal{E}, a \in S_m, b \in S_m\}, \\
\mathcal{B}_1(S_m) &= \{(a,b) | (a,b) \in \mathcal{E}, a \in S_m, b \notin S_m\}, \\
\mathcal{B}_2(S_m) &= \{(a,b) | (a,b) \in \mathcal{E}, a \notin S_m, b \in S_m\}, \\
\mathcal{B}(S_m) &= \mathcal{B}_1(S_m) \bigcup \mathcal{B}_2(S_m).
\end{aligned}
\tag{2}
$$

---

1. In what follows, we will assume that $j_m = i_m + L$ instead of $j_m = \min\{i_m + L, h - 1\}$. In other words, the length of the interval will always be L. However, all the arguments can be trivially extended to the case where the length of the interval is less than L.

At iteration m, the Range Swap algorithm moves from labeling f_m to f_{m+1} such that

$$Q(f_{m+1}, \mathbf{D}; \theta) \leq Q(f_m, \mathbf{D}; \theta),$$
$$f_{m+1}(a) \in I_m, \forall v_a \in \mathbf{v}(S_m),$$
$$f_{m+1}(a) = f_m(a), \forall v_a \in \mathbf{v} - \mathbf{v}(S_m),$$

where $\mathbf{v} - \mathbf{v}(S_m)$ denotes all the variables that are not present in the set $\mathbf{v}(S_m)$. The new labeling f_{m+1} is obtained by constructing a graph such that every st-cut on the graph corresponds to a labeling f of the random variables that satisfies:

$$f(a) \in I_m, \forall v_a \in \mathbf{v}(S_m),$$
$$f(a) = f_m(a), \forall v_a \in \mathbf{v} - \mathbf{v}(S_m).$$

The new labeling f_{m+1} is computed by solving for the minimum cost cut in this graph. We provide the details of the graph construction below.

5.1 Graph Construction

The Range Swap algorithm relies on a graph construction that is capable of exactly modeling arbitrary unary potentials and convex pairwise potentials. Such a graph construction was first proposed by Ishikawa (2003). As will be seen shortly, in this work we use a simpler graph construction (that does not require any *out-of-bounds* edges used in Ishikawa 2003). However, it is worth noting that the graph construction of Ishikawa (2003) may also be employed without affecting any property of the algorithm.

At each iteration of our algorithm, we are given an interval $I_m = [i_m + 1, j_m]$ of L labels (that is, $j_m = i_m + L$) where $d(L) = M$. We also have the current labeling f_m for all the random variables. We construct a directed weighted graph (with non-negative weights) $\mathcal{G}_m = \{\mathcal{V}_m, \mathcal{E}_m, c_m(\cdot, \cdot)\}$ such that for each $v_a \in \mathbf{v}(S_m)$, we define vertices $\{a_{i_m+1}, a_{i_m+2}, \cdots, a_{j_m}\} \in \mathcal{V}_m$. In addition, as is the case with every st-MINCUT problem, there are two additional vertices called terminals which we denote by s (the source) and t (the sink).

The edges $e \in \mathcal{E}_m$ with capacity (weight) $c_m(e)$ are defined to represent the following three types of potentials: (i) the unary potential $\theta_a(k)$ for random variable $v_a \in \mathbf{v}(S_m)$ taking the label k specified by an st-cut in the graph; (ii) the pairwise potential $\theta_{ab}(k, f_m(b))$ where $(a, b) \in \mathcal{B}_1(S_m)$ and the pairwise potential $\theta_{ab}(f_m(a), k)$ where $(a, b) \in \mathcal{B}_2(S_m)$, that is, pairwise potentials where one random variable is fixed to take its previous label; and (iii) the pairwise potential $\theta_{ab}(k, k')$ where $(a, b) \in \mathcal{A}(S_m)$, that is, pairwise potentials where no random variable is fixed to take its previous label. Note that all other potentials that specify the energy of the labeling are fixed during the iteration.

5.1.1 REPRESENTING UNARY POTENTIALS

For all random variables $v_a \in \mathbf{v}(S_m)$, we define the following edges that belong to the set $\mathcal{E}_m$:

- For all $k \in [i_m + 1, j_m)$, edges (a_k, a_{k+1}) have capacity $c_m(a_k, a_{k+1}) = \theta_a(k)$, that is, the cost of assigning label l_k to variable v_a.

- For all $k \in [i_m + 1, j_m)$, edges (a_{k+1}, a_k) have capacity $c_m(a_{k+1}, a_k) = \infty$.

- Edges (a_{j_m}, t) have capacity $c_m(a_{j_m}, t) = \theta_a(j_m)$.

- Edges (t, a_{j_m}) have capacity $c_m(t, a_{j_m}) = \infty$.

- Edges (s, a_{i_m+1}) have capacity $c_m(s, a_{i_m+1}) = \infty$.

- Edges (a_{i_m+1}, s) have capacity $c_m(a_{i_m+1}, s) = \infty$.

Fig. 2 shows the above edges together with their capacities for one random variable v_a. Note that there are two types of edges in the above set: (i) with finite capacity; and (ii) with infinite capacity. Any st-cut with finite cost contains only one of the finite capacity edges for each random variable v_a. This is because if an st-cut included more than one finite capacity edge, then by construction it must include at least one infinite capacity edge thereby making its cost infinite (Ishikawa, 2003). We interpret a finite cost st-cut as a relabeling of the random variables as follows:

$$f(a) = \begin{cases} k & \text{if } st\text{-cut includes edge } (a_k, a_{k+1}) \text{ where } k \in [i_m + 1, j_m), \\ j_m & \text{if } st\text{-cut includes edge } (a_{j_m}, t). \end{cases} \tag{3}$$

Note that the sum of the unary potentials for the labeling f is exactly equal to the cost of the st-cut over the edges defined above.
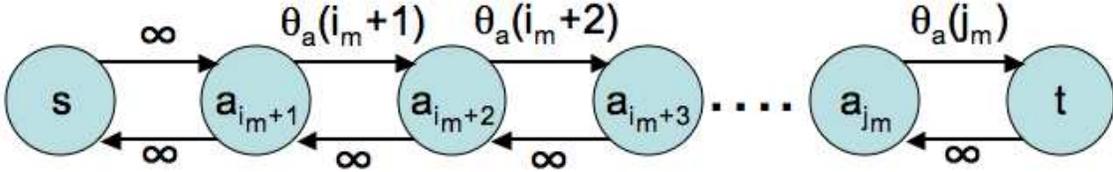

Figure 2: Part of the graph $\mathcal{G}_m$ containing the terminals and the vertices corresponding to the variable v_a. The edges that represent the unary potential of the new labeling are also shown.

5.1.2 REPRESENTING PAIRWISE POTENTIALS WITH ONE FIXED VARIABLE

We describe the case where $(a, b) \in \mathcal{B}_1(S_m)$. The other case where $(a, b) \in \mathcal{B}_2(S_m)$ can be handled similarly. Since $f_{m+1}(b)$ is fixed to $f_m(b)$, the pairwise potential $\theta_{ab}(i, f_{m+1}(b)) = \theta_{ab}(i, f_m(b))$ can be effectively treated as a unary potential of v_a. Hence, similar to unary potentials, it can be formulated using the following edge in set $\mathcal{E}_m$:

- For all $k \in [i_m + 1, j_m)$, edges (a_k, a_{k+1}) have capacity $c_m(a_k, a_{k+1}) = \theta_{ab}(k, f_m(b))$, that is, the cost of assigning label l_k to variable v_a and keeping the label of v_b fixed to $f_m(b)$.

- For all $k \in [i_m + 1, j_m)$, edges (a_{k+1}, a_k) have capacity $c_m(a_{k+1}, a_k) = \infty$.

- Edges (a_{j_m}, t) have capacity $c_m(a_{j_m}, t) = \theta_{ab}(j_m, f_m(b))$.

- Edges (t, a_{j_m}) have capacity $c_m(t, a_{j_m}) = \infty$.

- Edges (s, a_{i_m+1}) have capacity $c_m(s, a_{i_m+1}) = \infty$.

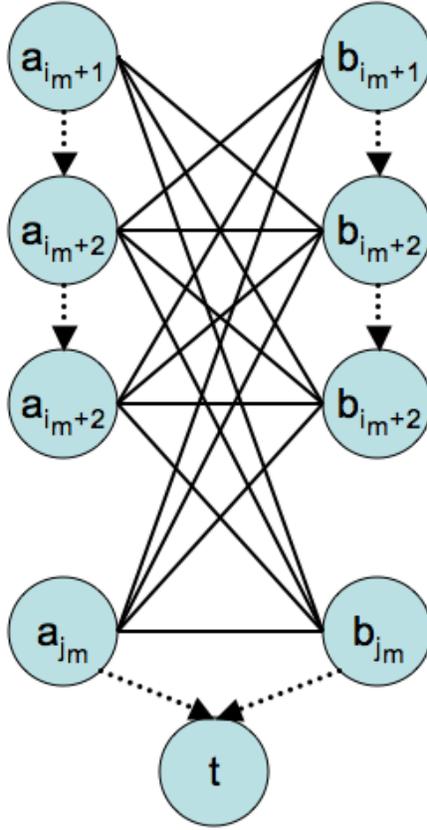- Edges (a_{i_m+1}, s) have capacity $c_m(a_{i_m+1}, s) = \infty$.

Figure 3: Edges that are used to represent the pairwise potentials of two neighboring random variables v_a and v_b such that $(a,b) \in \mathcal{A}(S_m)$ are shown. Undirected edges indicate that there are directed edges in both directions with equal capacity (as given by Equation 4). Directed dashed edges, with capacities shown in Equation (5), are added to ensure that the graph models the convex pairwise potentials correctly.

5.1.3 REPRESENTING PAIRWISE POTENTIALS WITH NO FIXED VARIABLES

For all random variables v_a and v_b such that $(a,b) \in \mathcal{A}(S_m)$, we define edges $(a_k, b_{k'}) \in \mathcal{E}_m$ where either one or both of k and k' belong to the set $(i_m + 1, j_m]$ (that is, at least one of them is not $i_m + 1$). The capacity of these edges is given by

$$c_m(a_k, b_{k'}) = \frac{w_{ab}}{2} \left(d(k - k' + 1) - 2d(k - k') + d(k - k' - 1) \right). \tag{4}$$

The RHS of the above equation is guaranteed to be non-negative due to the fact that $w_{ab} \geq 0$ and $d(\cdot)$ is convex. It is worth noting that, for the special cases when $d(\cdot)$ is linear or quadratic, the above capacity has a simple form. Specifically, when $d(\cdot)$ is linear the above capacity is equal to w_{ab} if $k = k'$ and 0 otherwise. When $d(\cdot)$ is quadratic the above capacity is a constant w_{ab} for all values of

k and k'. In addition to the edges described in Equation (4), we also specify the following edges:

$$c_m(a_k, a_{k+1}) = \frac{w_{ab}}{2}\left(d(L-k+i_m)+d(k-i_m)\right), \forall (a,b) \in \mathcal{E}, k \in [i_m+1, j_m),$$

$$c_m(b_{k'}, b_{k'+1}) = \frac{w_{ab}}{2}\left(d(L-k'+i_m)+d(k'-i_m)\right), \forall (a,b) \in \mathcal{E}, k' \in [i_m+1, j_m),$$

$$c_m(a_{j_m}, t) = c_m(b_{j_m}, t) = \frac{w_{ab}}{2}d(L), \forall (a,b) \in \mathcal{E}. \tag{5}$$

Fig. 3 provides an illustration of the above edges. The following Lemma shows that these edges model convex pairwise potentials exactly (up to an additive constant).

Lemma 1: For the capacities defined in Equations (4) and (5), the cost of the *st*-cut which includes the edges (a_k, a_{k+1}) and $(b_{k'}, b_{k'+1})$ (that is, v_a and v_b take labels l_k and $l_{k'}$ respectively) is given by $w_{ab}d(k-k') + \kappa_{ab}$, where the constant $\kappa_{ab} = w_{ab}d(L)$ (Proof in Appendix B).

This completes our graph construction. Given the graph $\mathcal{G}_m$ we solve the *st*-MINCUT problem, which provides us with a labeling f_{m+1} (using Equation (3)). We note that, since the cost of the *st*-cut exactly models the convex pairwise potential plus a constant, the above graph (together with the edges representing unary potentials) can be used to find the exact MAP estimate of the random field with convex pairwise potentials. In other words, ours is a somewhat modified, easy to follow graph construction for the method of Ishikawa (2003).

5.2 Generalizing Range Swap

In the previous subsection, we had assumed that the length of the interval L was chosen such that $d(L) = M$. We now relax this assumption such that $d(L) \geq M$. In this case, we define the set S_m such that

$$S_m = \{a|f_m(a) \in I_m, d(f_m(a), f_m(b)) \leq M, \forall (a,b) \in \mathcal{E}, f_m(b) \in I_m\}.$$

In other words, S_m consists of those random variables whose current label belongs to the interval I_m and whose pairwise potential with all its neighboring random variables v_b such that $f_m(b) \in I_m$ lies in the convex part of the truncated convex model. Using S_m the subset of random variables $\mathbf{v}(S_m)$ and the subset of edges $\mathcal{A}(S_m)$, $\mathcal{B}_1(S_m)$, $\mathcal{B}_2(S_m)$ and $\mathcal{B}(S_m)$ are defined as in Equation (2). The graph over which the *st*-MINCUT is performed is constructed as described in the previous subsection. As will be seen in § 5.4, the above definition of S_m would be useful in proving that the Range Swap algorithm monotonically improves the energy of the labeling from one iteration to the next.

5.3 Properties of the Graph

The following properties relating an *st*-cut with the corresponding labeling f hold true for the graph construction described in the previous subsection.

Property 1: The cost of the *st*-cut exactly represents the sum of the unary potentials for all variables in $\mathbf{v}(S_m)$, that is, $\sum_{v_a \in \mathbf{v}(S_m)} \theta_a(f(a))$.

Property 2: For $(a,b) \in \mathcal{B}_1(S_m)$, the cost of the *st*-cut exactly represents the pairwise potential $\theta_{ab}(f(a), f_m(b))$. Similarly, for $(a,b) \in \mathcal{B}_2(S_m)$, the cost of the *st*-cut exactly represents the pairwise potential $\theta_{ab}(f_m(a), f(b))$.

Property 3: For $(a,b) \in \mathcal{A}(S_m)$, if $f(a) \in I_m$ and $f(b) \in I_m$ such that

$$d(f(a) - f(b)) \leq M,$$

then the cost of the *st*-cut exactly represents the pairwise potential $\theta_{ab}(f(a), f(b))$ plus a constant κ_{ab}, that is,

$$w_{ab}d(f(a) - f(b)) + \kappa_{ab}.$$

This property follows directly from Lemma 1.

Property 4: For $(a, b) \in \mathcal{A}(S_m)$, if $f(a) \in I_m$ and $f(b) \in I_m$ such that

$$d(f(a) - f(b)) > M,$$

then the cost of the *st*-cut incorrectly represents the pairwise potential $\theta_{ab}(f(a), f(b))$, being

$$w_{ab}d(f(a) - f(b)) + \kappa_{ab},$$

which is an overestimation of the correct value (that is, $w_{ab}M$ plus the constant κ_{ab}). This follows from the fact that our graph construction overestimates the truncation part by the convex function $w_{ab}d(\cdot)$.

In summary, property 1 tells us that the cost of the *st*-cut exactly models the sum of the unary potentials. Properties 2 and 3 specify the cases where the cost of the *st*-cut exactly models the pairwise potentials, while property 4 specifies the remaining case where the cost of the *st*-cut overestimates the pairwise potentials. Since the potentials are either modeled exactly or are overestimated, it follows that the energy of the labeling f_{m+1} is less than or equal to the cost of the *st*-MINCUT on $\mathcal{G}_m$. The only free parameter in the Range Swap algorithm is the length of the interval L. Next, we discuss how to choose the value of this parameter.

5.4 Length of the Interval

We begin by considering the case when L satisfies $d(L) = M$. Note that in this case, property 4 no longer needs to be considered. This implies that the cost of the *st*-cut exactly models the energy of the corresponding labeling. Hence, the *st*-MINCUT provides the optimal move f_{m+1}. Next, we consider the case when the length of the interval satisfies $d(L) > M$. We show that this interval provides a labeling that is at least as good as the labeling obtained by considering any of its subsets for which the optimal move can be computed. Formally, let f_{m+1} be the labeling obtained by using an interval of length L such that $d(L) > M$ and let f'_{m+1} be the labeling obtained by using a subset of the interval of length L' such that $d(L') = M$. Then the following holds true.

Observation 1: The energy of f_{m+1} is less than or equal to the energy of f'_{m+1}.

Proof: When we use the interval of length L, one of the cuts in the graph would correspond to f'_{m+1}. Since $d(L') = M$, it follows that the cost of the cut would be equal to $Q(f'_{m+1}, \mathbf{D}; \theta)$. Furthermore, the cost of the cut corresponding to f_{m+1} is at least equal to $Q(f_{m+1}, \mathbf{D}; \theta)$. Using the fact that f_{m+1} corresponds to the minimum cost cut, we see that

$$Q(f_{m+1}, \mathbf{D}; \theta) \leq Q(f'_{m+1}, \mathbf{D}; \theta).$$

∎

The above observation shows that we do not lose any accuracy by considering non-optimal moves on large intervals (compared to optimal moves on smaller subsets of the interval). However, the larger the value of L the bigger the corresponding graph on which we need to compute the *st*-MINCUT. Thus, in practice the value of L should be chosen according to the available computational resources.

5.5 Analysis of Range Swap

Regardless of whether $d(L) = M$ (that is, the Range Swap algorithm described in § 5.1) or $d(L) > M$ (its generalization described in § 5.2), it is worth noting that the corresponding graph construction ensures that the cut corresponding to the labeling f_m exactly models the energy $Q(f_m, \mathbf{D}; \theta)$ up to a constant. This implies that the energy of the new labeling f_{m+1} is less than or equal to the energy of f_m, that is,

$$Q(f_{m+1}, \mathbf{D}; \theta) \leq Q(f_m, \mathbf{D}; \theta).$$

This follows from the fact that the cost of the st-MINCUT is less than or equal to the energy of the labeling f_m but is greater than or equal to the energy of f_{m+1}. In other words, the Range Swap algorithm monotonically improves the energy of the labeling from one iteration to the next.

It is worth noting that, unlike previous move making algorithms, Range Swap is not guaranteed to compute the optimal move other than in the special case when $d(L) = M$ (where $L = j_m - i_m$ is the length of the interval). In other words, for the case where $d(L) > M$, if in the mth iteration we move from label f_m to f_{m+1} then it is possible that there exists another labeling f'_{m+1} such that

$$Q(f'_{m+1}, \mathbf{D}; \theta) \leq Q(f_{m+1}, \mathbf{D}; \theta),$$
$$f'_{m+1}(a) \in I_m, \forall v_a \in \mathbf{v}(S_m),$$
$$f'_{m+1}(a) = f_m(a), \forall v_a \in \mathbf{v} - \mathbf{v}(S_m).$$

This is due to the fact that the graph construction overestimates certain pairwise potentials (see Property 4). However, as Observation 1 shows, the improvement in the energy obtained by a (potentially non-optimal) move when $d(L) > M$ is at least as much as the improvement obtained by the optimal move when $d(L) = M$.

6. The Range Expansion Algorithm

Range Expansion is a suitable modification of the α-expansion algorithm of Boykov et al. (2001) for truncated convex models. Unlike Range Swap, at an iteration m it considers all the random variables v_a regardless of whether their current labeling $f_m(a)$ lies in the interval I_m. It provides the option for each random variable v_a to either retain its old label $f_m(a)$ or change its label to $f_{m+1}(a) \in I_m$. Formally, the Range Expansion algorithm moves from labeling f_m to f_{m+1} such that

$$Q(f_{m+1}, \mathbf{D}; \theta) \leq Q(f_m, \mathbf{D}; \theta),$$
$$f_{m+1}(a) = f_m(a) \text{ OR } f_{m+1}(a) \in I_m, \forall v_a \in \mathbf{v}.$$

Similar to Range Swap, Range Expansion does not compute the optimal move at each iteration. In other words, if in the mth iteration we move from label f_m to f_{m+1} then it is possible that there exists another labeling f'_{m+1} such that

$$Q(f'_{m+1}, \mathbf{D}; \theta) < Q(f_{m+1}, \mathbf{D}; \theta),$$
$$f'_{m+1}(a) = f_m(a) \text{ OR } f'_{m+1}(a) \in I_m, \forall v_a \in \mathbf{v}.$$

However, our analysis in § 6.3 shows that we will still be able to reduce the energy sufficiently at each iteration so as to obtain the best known multiplicative bounds upon convergence. As in the

case of Range Swap, we move from labeling f_m to f_{m+1} by constructing a graph such that every st-cut on the graph corresponds to a labeling f of the random variables that satisfies:

$$f(a) = f_m(a) \text{ OR } f(a) \in I_m, \forall v_a \in \mathbf{v}.$$

The new labeling f_{m+1} is obtained in two steps: (i) we obtain a labeling f that corresponds to the st-MINCUT on our graph; and (ii) we choose the new labeling f_{m+1} as

$$f_{m+1} = \begin{cases} f & \text{if} & Q(f, \mathbf{D}; \theta) \leq Q(f_m, \mathbf{D}; \theta), \\ f_m & \text{otherwise.} \end{cases} \tag{6}$$

Note that, unlike Range Swap, step (ii) is required in Range Expansion since the labeling f obtained in step (i) may have greater energy than f_m. This is due to the approximations involved in the graph construction described below.

6.1 Graph Construction

We construct a directed weighted graph (with non-negative weights) $G_m = \{V_m, E_m, c_m(\cdot, \cdot)\}$ such that V_m contains the source s, the sink t and the vertices $\{a_{i_m+1}, a_{i_m+2}, \cdots, a_{j_m}\}$ for each random variable $v_a \in \mathbf{v}$. The edges $e \in E_m$ with capacity $c_m(e)$ are of two types: (i) those that represent the unary potentials of a labeling corresponding to an st-cut in the graph and; (ii) those that represent the pairwise potentials of the labeling.
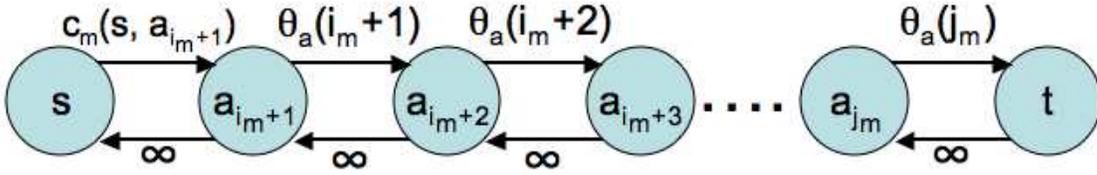

Figure 4: Part of the graph G_m containing the terminals and the vertices corresponding to the variable v_a. The edges that represent the unary potential of the new labeling are also shown. The term $c_m(s, a_{i_m+1})$ is defined in Equation (7).

6.1.1 REPRESENTING UNARY POTENTIALS

The unary potentials are represented in a similar manner to the graph construction used in Range Swap. The notable difference is that now we have to model the unary potential for the case when a variable v_a retains its old label that does not lie in the interval I_m. To this end, we change the capacity of the edge (s, a_{i_m+1}) to

$$c_m(s, a_{i_m+1}) = \begin{cases} \theta_a(f_m(a)) & \text{if} & f_m(a) \notin I_m, \\ \infty & \text{otherwise.} \end{cases} \tag{7}$$

Fig. 4 shows all the edges specified for representing the unary potential of one random variable v_a. We interpret a finite cost st-cut as a relabeling of the random variables as follows:

$$f(a) = \begin{cases} k & \text{if } st\text{-cut includes edge } (a_k, a_{k+1}) \text{ where } k \in [i_m+1, j_m), \\ j_m & \text{if } st\text{-cut includes edge } (a_{j_m}, t), \\ f_m(a) & \text{if } st\text{-cut includes edge } (s, a_{i_m+1}). \end{cases} \tag{8}$$

Note that the sum of the unary potentials for the labeling f is exactly equal to the cost of the st-cut over the edges defined above.

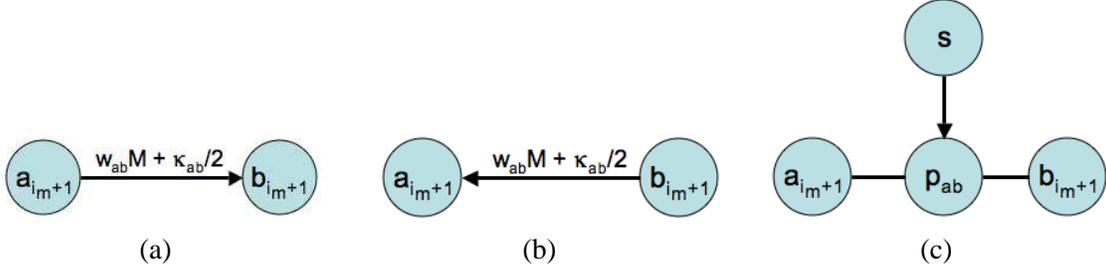(a)   (b)   (c)

Figure 5: Additional edges that are added to the graph shown in Fig. 3 for representing pairwise potentials. **(a)** When $f_m(a) \in I_m$ and $f_m(b) \notin I_m$. Here, $\kappa_{ab} = w_{ab}d(L)$. **(b)** When $f_m(a) \notin I_m$ and $f_m(b) \in I_m$. **(c)** When $f_m(a) \notin I_m$ and $f_m(b) \notin I_m$. Undirected edges indicate the presence of opposing edges with equal capacity. The capacities of all five edges are specified in Equation (9).

6.1.2 REPRESENTING PAIRWISE POTENTIALS

For each pair of neighboring random variables $(a,b) \in \mathcal{E}$ we will use the edges defined for the graph of Range Swap for representing pairwise potentials, that is, all the edges shown in Fig. 3. However, we also have to consider the cases where at least one of the neighboring random variables retains its previous label and that label is not present in the interval I_m. In order to model these cases, we incorporate the following additional edges:

- If $f_m(a) \in I_m$ and $f_m(b) \notin I_m$ then we add an edge (a_{i_m+1}, b_{i_m+1}) with capacity $w_{ab}M + \kappa_{ab}/2$ (see Fig. 5(a)).

- If $f_m(a) \notin I_m$ and $f_m(b) \in I_m$ then we add an edge (b_{i_m+1}, a_{i_m+1}) with capacity $w_{ab}M + \kappa_{ab}/2$ (see Fig. 5(b)).

- If $f_m(a) \notin I_m$ and $f_m(b) \notin I_m$, we introduce a new vertex p_{ab}.[2] Using this vertex p_{ab}, five edges are defined with the following capacities (see Fig. 5(c)):

$$c_m(a_{i_m+1}, p_{ab}) = c_m(p_{ab}, a_{i_m+1}) = w_{ab}M + \kappa_{ab}/2,$$
$$c_m(b_{i_m+1}, p_{ab}) = c_m(p_{ab}, b_{i_m+1}) = w_{ab}M + \kappa_{ab}/2,$$
$$c_m(s, p_{ab}) = \theta_{ab}(f_m(a), f_m(b)) + \kappa_{ab}.$$

This completes our graph construction. Given the graph $\mathcal{G}_m$ we solve the st-MINCUT problem, which provides us with a labeling f as described in Equation (8). The new labeling f_{m+1} is obtained using Equation (6).

---

2. We note here that an equivalent graph can be constructed without adding the vertex p_{ab} using the method of Schlesinger and Flach (2006). However, the vertex p_{ab} helps make the analysis easier.

6.2 Properties of the Graph

We now describe the properties of the above graph construction, with the aim of facilitating the analysis of our algorithm for the case of truncated linear and truncated quadratic models.

Property 5: The cost of the *st*-cut exactly represents the sum of the unary potentials associated with the corresponding labeling f, that is, $\sum_{v_a \in \mathbf{v}} \theta_a(f(a))$.

Property 6: For $(a,b) \in \mathcal{E}$, if $f(a) = f_m(a) \notin I_m$ and $f(b) = f_m(b) \notin I_m$ then the cost of the *st*-cut exactly represents the pairwise potential $\theta_{ab}(f(a), f(b))$ plus a constant κ_{ab}. This is due to the fact that the *st*-cut contains the edge (s, p_{ab}) whose capacity is $\theta_{ab}(f_m(a), f_m(b)) + \kappa_{ab}$. Note that in this case p_{ab} belongs to the partition containing the sink t. This can be easily verified by observing that the cost of the *st*-cut would increase if p_{ab} belonged to the partition containing the source s (since this would include edge (p_{ab}, a_{i_m+1}) and (p_{ab}, b_{i_m+1}) in the *st*-cut).

Property 7: For $(a,b) \in \mathcal{E}$, if $f(a) \in I_m$ and $f(b) \in I_m$ such that

$$d(f(a) - f(b)) \leq M,$$

then the cost of the *st*-cut exactly represents the pairwise potential $\theta_{ab}(f(a), f(b))$ plus a constant κ_{ab}, that is,

$$w_{ab}d(f(a) - f(b)) + \kappa_{ab}.$$

This follows from the fact that in this case the pairwise potential lies in the convex part of the truncated convex model, which is modeled exactly (see Lemma 1).

Property 8: For $(a,b) \in \mathcal{E}$, if $f(a) \in I_m$ and $f(b) \in I_m$ such that

$$d(f(a) - f(b)) > M,$$

then the cost of the *st*-cut incorrectly represents the pairwise potential $\theta_{ab}(f(a), f(b))$, being

$$w_{ab}d(f(a) - f(b)) + \kappa_{ab},$$

which is an overestimation of the correct value (that is, $w_{ab}M$ plus the constant κ_{ab}). This follows from the fact that our graph construction overestimates the truncation part by the convex function $w_{ab}d(\cdot)$ (see Lemma 1).

Property 9: For $(a,b) \in \mathcal{E}$, if $f(a) \in I_m$ and $f(b) = f_m(b) \notin I_m$ then the cost of the *st*-cut incorrectly represents the pairwise potential $\theta_{ab}(f(a), f(b))$, being

$$w_{ab}d(f(a) - (i_m + 1)) + w_{ab}\hat{d}(f(a) - (i_m + 1)) + w_{ab}M + \kappa_{ab}, \tag{9}$$

where $\hat{d}(\cdot)$ denotes the following function:

$$\hat{d}(x) = d(x+1) - d(x) - d(1) + \frac{d(0)}{2}, \forall x \geq 0. \tag{10}$$

Note that $\hat{d}(\cdot)$ is only defined for a non-negative argument. Clearly, the argument of $\hat{d}(\cdot)$ in Equation (9) is non-negative since $f(a) \in [i_m + 1, j_m]$. The function $\hat{d}(x) = 0$ when $d(\cdot)$ is a linear metric and $\hat{d}(x) = 2x$ when $d(\cdot)$ is the quadratic semi-metric.

Similarly, if $f(a) = f_m(a) \notin I_m$ and $f(b) \in I_m$ then the cost of the *st*-cut incorrectly represents the pairwise potential $\theta_{ab}(f(a), f(b))$, being

$$w_{ab}d(f(b) - (i_m + 1)) + w_{ab}\hat{d}(f(b) - (i_m + 1)) + w_{ab}M + \kappa_{ab}.$$

The above property can be shown to be true using the following Lemma.

Lemma 2: For the graph described in § 6.1, property 9 holds true (Proof in Appendix C).

In summary, property 5 tells us that the cost of the *st*-cut exactly models the sum of the unary potentials. Properties 6 and 7 specify the cases where the cost of the *st*-cut exactly models the pairwise potentials, while properties 8 and 9 specify the remaining cases where the cost of the *st*-cut overestimates the pairwise potentials. In other words, the energy of the labeling f, and hence the energy of f_{m+1}, is less than or equal to the cost of the *st*-MINCUT on $\mathcal{G}_m$.

Note that our graph construction is similar to that of Gupta and Tardos (2000) with two notable exceptions: (i) we can handle any general truncated convex model and not just truncated linear as in the case of Gupta and Tardos (2000); and (ii) we have the freedom to choose the value of L, while Gupta and Tardos (2000) fixed this value to M. A logical choice would be to use that value of L that minimizes the worst case multiplicative bound for a particular class of problems. The following analysis obtains the desired value of L for both the truncated linear and the truncated quadratic models. Our worst case multiplicative bounds are exactly those achieved by the LP relaxation (see Chekuri et al., 2005).

6.3 Multiplicative Bounds

In order to obtain multiplicative bounds for the Range Expansion algorithm, we will make use of the fact that the algorithm only terminates once we are unable to reduce the energy for any interval I_m. In other words, we stop once we have reached the local minimum of the large neighborhood defined by the intervals. We exploit this fact in the following manner. First, we establish a lower bound on how much the energy is reduced for a given interval (see Lemma 3 below). To this end, we extensively use the properties of the graph described in the previous subsection. As our final labeling f is a local minimum over the intervals, it follows that once the algorithm terminates the above mentioned lower bound will be less than or equal to zero (otherwise it would be possible to reduce the energy further). This observation provides us with an expression for the upper bound of the energy of f. Next, we simplify this expression for both truncated linear metric (see Theorem 2) and truncated quadratic semi-metric (see Theorem 3) and show that our bounds match those of the LP relaxation.

Before we proceed with the details, we require the following definitions. Let $r \in [0, L-1]$ be a uniformly distributed random integer. Using r we define the following set of intervals

$$I_r = \{[0,r], [r+1, r+L], [r+L+1, r+2L], \cdots, [., h-1]]\},$$

where $h = |\mathbf{l}|$ is the total number of labels associated with the MRF. We denote an optimal labeling of the MRF by f^*. Given such a labeling f^* and an interval $I_m = [i_m + 1, j_m] \in I_r$, we define the following sets:

$$\mathbf{v}(f^*, I_m) = \{v_a | v_a \in \mathbf{v}, f^*(a) \in I_m\},$$
$$\mathcal{A}(f^*, I_m) = \{(a,b) | (a,b) \in \mathcal{E}, f^*(a) \in I_m, f^*(b) \in I_m\},$$
$$\mathcal{B}_1(f^*, I_m) = \{(a,b) | (a,b) \in \mathcal{E}, f^*(a) \in I_m, f^*(b) \notin I_m\},$$
$$\mathcal{B}_2(f^*, I_m) = \{(a,b) | (a,b) \in \mathcal{E}, f^*(a) \notin I_m, f^*(b) \in I_m\},$$
$$\mathcal{B}(f^*, I_m) = \mathcal{B}_1(f^*, I_m) \cup \mathcal{B}_2(f^*, I_m).$$

In other words, $\mathbf{v}(f^*, I_m)$ contains all the random variables that take an optimal labeling in I_m, $\mathcal{A}(f^*, I_m)$ contains the set of all edges in the graphical model of the MRF whose endpoints take an

optimal labeling in the interval I_m, and $\mathcal{B}(f^*, I_m)$ contains edges where only one endpoint takes an optimal labeling in I_m.

Clearly, the following equation holds true:

$$\sum_{v_a \in \mathbf{v}} \theta_a(f^*(a)) = \sum_{I_m \in I_r} \sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f^*(a)), \tag{11}$$

since $f^*(a)$ belongs to exactly one interval in I_r for all $v_a \in \mathbf{v}$. In order to make the analysis less cluttered, we introduce the following shorthand notation for some terms:

- For $(a, b) \in \mathcal{A}(f^*, I_m)$, we denote $w_{ab} d(f^*(a) - f^*(b))$ by e_{ab}^m.

- For $(a, b) \in \mathcal{B}_1(f^*, I_m)$, we denote $w_{ab} d(f^*(a) - (i_m + 1)) + w_{ab} \hat{d}(f^*(a) - (i_m + 1)) + w_{ab} M$ by e_a^m.

- For $(a, b) \in \mathcal{B}_2(f^*, I_m)$, we denote $w_{ab} d(f^*(b) - (i_m + 1)) + w_{ab} \hat{d}(f^*(b) - (i_m + 1)) + w_{ab} M$ by e_b^m.

We are now ready to prove our main results, starting with the following Lemma.

Lemma 3: At an iteration of our algorithm, given the current labeling f_m and an interval $I_m = [i_m + 1, j_m]$, the new labeling f_{m+1} obtained by solving the st-MINCUT problem reduces the energy by at least the following:

$$\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f_m(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m) \cup \mathcal{B}(f^*, I_m)} \theta_{ab}(f_m(a), f_m(b))$$

$$- \left(\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f^*(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right).$$

Here f^* refers to an optimal labeling for the given MRF (Proof in Appendix D).

Let f be the final labeling obtained using our algorithm. Since f is a local optimum with respect to all intervals I_m, it follows that the above term should be non-positive for all I_m (otherwise the energy could be further reduced thereby contradicting the fact that f is a local optimum labeling). In other words,

$$\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m) \cup \mathcal{B}(f^*, I_m)} \theta_{ab}(f(a), f(b))$$

$$\leq \left(\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f^*(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right), \forall I_m.$$

We sum the above inequality over all $I_m \in I_r$. The summation of the LHS is at least $Q(f, \mathbf{D}; \theta)$. Furthermore, using Equation (11), the summation of the above inequality can be written as

$$Q(f, \mathbf{D}; \theta) \leq \sum_{v_a \in \mathbf{v}} \theta_a(f^*(a)) +$$

$$\sum_{I_m \in I_r} \left(\sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right).$$

We now take the expectation of the above inequality over the uniformly distributed random integer $r \in [0, L-1]$. The LHS of the inequality and the first term on the RHS (that is, $\sum \theta_a(f^*(a))$) are constants with respect to r. Hence, we get

$$
Q(f, \mathbf{D}; \theta) \leq \sum_{v_a \in \mathbf{v}} \theta_a(f^*(a)) +
$$

$$
\frac{1}{L} \sum_r \sum_{I_m \in I_r} \left(\sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right). \tag{12}
$$

We conclude by observing that this is the same bound that is obtained by the LP relaxation. Thus, using the analysis of Chekuri et al. (2005) we obtain the following results.

Lemma 4: When $d(\cdot)$ is linear, that is, $d(x) = |x|$, the following inequality holds true:

$$
\frac{1}{L} \sum_r \sum_{I_m \in I_r} \left(\sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right)
$$

$$
\leq \left(2 + \max\left\{ \frac{2M}{L}, \frac{L}{M} \right\} \right) \sum_{(a,b) \in \mathcal{E}} \theta_{ab}(f^*(a) f^*(b)).
$$

(Proof in Appendix E).

Theorem 2: For the truncated linear metric, our algorithm obtains a multiplicative bound of $2 + \sqrt{2}$ using $L = \sqrt{2}M$.

The proof of the above theorem follows by substituting $L = \sqrt{2}M$ in the above inequality and simplifying inequality (12). Note that this bound is better than those obtained by α-expansion (Boykov et al., 2001) ($2M$) and its generalization (Gupta and Tardos, 2000) (4). In fact, the bound of Gupta and Tardos (2000) can be obtained directly from the above analysis by using the non-optimal assignment of $L = M$.

Similarly, using Theorem 4 of Chekuri et al. (2005), we obtain the following multiplicative bound for the truncated quadratic semi-metric.

Theorem 3: For the truncated quadratic semi-metric, our algorithm obtains a multiplicative bound of $O(\sqrt{M})$ using $L = \sqrt{M}$.

Note that both α-expansion and the approach of Gupta and Tardos provide no bounds for the above case. The primal-dual method of Komodakis and Tziritas (2007) obtains a bound of $2M$, which is clearly inferior to our guarantees. Finally, we note that a slight modification of Theorem 3.7 of Gupta and Tardos (2000) shows that the above guarantees can be obtained in a polynomial number of iterations. Since each iteration itself is of polynomial complexity, it follows that the Range Expansion algorithm provides LP multiplicative bounds on polynomial time.

Theorem 4: If the Range Expansion algorithm is run for $O(h/L)(\log Q(f_1, \mathbf{D}; \theta) + \log \varepsilon^{-1})$ iterations (where f_1 is the initial labeling, and $\varepsilon > 0$), then the expected value of the energy would be at most $(2 + \sqrt{2} + \varepsilon)Q(f^*, \mathbf{D}; \theta)$ for the truncated linear metric and $(O(\sqrt{M}) + \varepsilon)Q(f^*, \mathbf{D}; \theta)$ for the truncated quadratic semi-metric (where f^* is an optimal labeling).

Although theoretically interesting, the practical implications of this result are minimal since in most scenarios we will be able to run our methods for a sufficient number of iterations so as to end up in a local minimum over all intervals I_m. For instance, in all our experiments we reached a local minimum in less than 5 iterations.

7. Experiments

We tested both our range move algorithms using both synthetic and standard real data, and compared it with several state of the art methods. We do not include a comparison with interior point algorithms for the primal LP due to their high computation cost.

7.1 Synthetic Data

We begin with experiments that use synthetically generated data. In the following, we provide the details of the experimental setup and present the results obtained.

7.1.1 EXPERIMENTAL SETUP

We used 100 random fields for both the truncated linear and truncated quadratic models. The variables $\mathbf{v}$ and neighborhood relationship $\mathcal{E}$ of the random fields described a 4-connected grid graph of size 50×50. Note that 4-connected grid graphs are widely used to model several problems in computer vision (Szeliski et al., 2008). Each variable was allowed to take one of 20 possible labels, that is, $\mathbf{l} = \{l_0, l_1, \cdots, l_{19}\}$. The parameters of the random field were generated randomly. Specifically, the unary potentials $\theta_a(i)$ were sampled uniformly from the interval $[0, 10]$ while the weights w_{ab}, which determine the pairwise potentials, were sampled uniformly from $[0, 5]$. The parameter M was also chosen randomly while taking care that $d(5) \leq M \leq d(10)$.

7.1.2 RESULTS

Fig. 6 shows the results obtained by our methods and four other state of the art algorithms: $\alpha\beta$-swap, α-expansion, BP and TRW-S. We used publicly available code for all previously proposed approaches.[3] As can be seen from the figure, the most accurate move making approaches are the methods proposed in this paper. As expected, both our algorithms are slower than $\alpha\beta$-swap and α-expansion (since each iteration computes an *st*-MINCUT on a larger graph). However, they are faster than TRW-S, which attempts to minimize the LP relaxation, and BP. We note here that our implementation does not use any clever tricks to speed up the max-flow algorithm (such as those described by Alahari et al., 2008) that can potentially decrease the running time by orders of magnitude.

7.2 Real Data - Stereo Reconstruction

Given two *epipolar rectified* images $\mathbf{D}_1$ and $\mathbf{D}_2$ of the same scene, the problem of stereo reconstruction is to obtain a correspondence between the pixels of the images. This problem can be modeled using a random field whose variables correspond to pixels of one image (say $\mathbf{D}_1$) and take labels from a set of *disparities* $\mathbf{l} = \{0, 1, \cdots, h-1\}$. A disparity value i for a random variable a denoting pixel (x, y) in $\mathbf{D}_1$ indicates that its corresponding pixel lies in location $(x+i, y)$ in the second image.

For the above random field formulation, the unary potentials were obtained using the method described by Birchfield and Tomasi (1998) and were truncated at 15. As is typically the case, we chose the neighborhood relationship $\mathcal{E}$ to define a 4-neighborhood grid graph. The number of

---

3. When using α-expansion with the truncated quadratic semi-metric, all edges with negative capacities in the graph construction were removed, similar to the experiments in Szeliski et al. (2008).
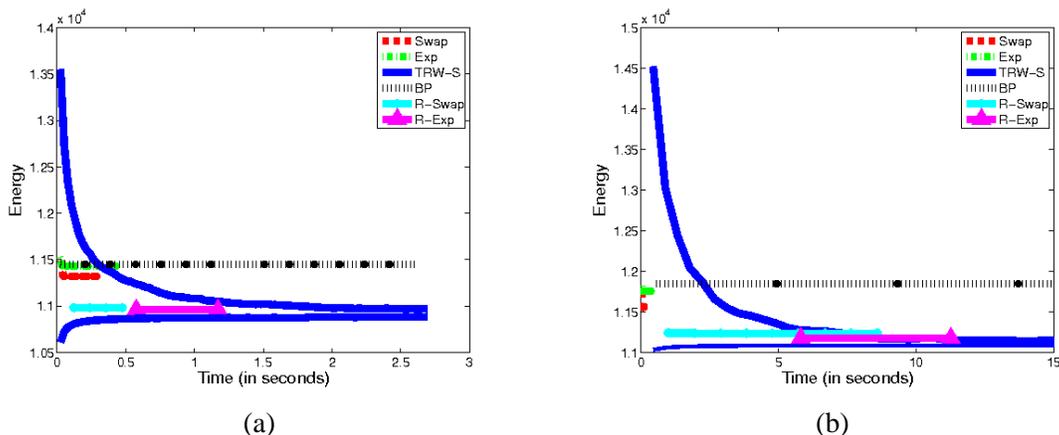
Figure 6: Results of the synthetic experiment. **(a)** Truncated linear metric. **(b)** Truncated quadratic semi-metric. The x-axis shows the time taken in seconds. The y-axis shows the average energy obtained over all 100 random fields using the six algorithms. The lower blue curve is the value of the dual obtained by TRW-S. In both the cases, our methods provide more accurate solutions than previous move making algorithms ($\alpha\beta$-swap and α-expansion) and are faster than message passing approaches (TRW-S and BP). The labelings obtained by our methods always have a lower energy than those obtained by BP and are comparable to the energy obtained using TRW-S.

disparities h was set to 20. We experimented using the following truncated convex potentials:

$$\theta_{ab}(i,j) = 50\min\{|i-j|, 10\},$$
$$\theta_{ab}(i,j) = 50\min\{(i-j)^2, 100\}.$$

The above form of pairwise potentials encourage neighboring pixels to take similar disparity values, which corresponds to our expectations of finding smooth surfaces in natural images. Truncation of pairwise potentials is essential to avoid over smoothing, as observed in Boykov et al. (2001). Note that using spatially varying weights w_{ab} provides better results. However, the main aim of this experiment is to demonstrate the accuracy and speed of our approach and not to design the best possible energy. Fig. 7 shows the results obtained using various algorithms when using the truncated linear metric on a standard stereo pair (Tsukuba). Table 4 provides the value of the energy and the total time taken by all the approaches for three stereo pairs. Similar to the synthetic experiments, the range move algorithms provide accurate solutions while taking less time than TRW-S and BP. Range expansion does marginally better than range swap but is computationally more expensive.

8. Concluding Remarks

Summary. We proposed the Range Swap and Range Expansion algorithms for obtaining an approximate MAP estimate of discrete random fields with truncated convex pairwise potentials. Our methods consider a range of labels at each iteration and hence, explore a larger search space compared to previous *st*-MINCUT based approaches. Due to the use of only the *st*-MINCUT problem
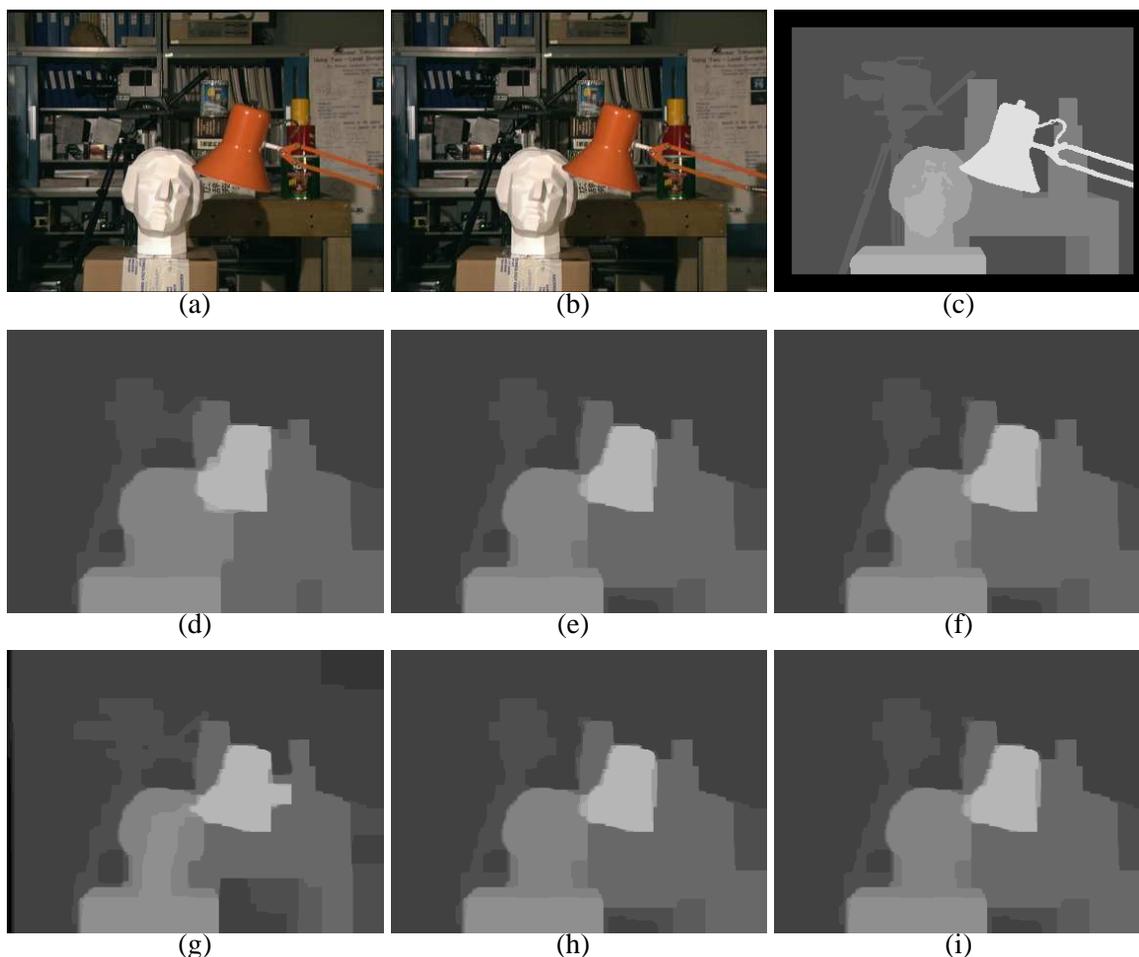
Figure 7: Tsukuba stereo pair. **(a)** First image. **(b)** Second image. **(c)** Ground truth disparity map. **(d)-(i)** Results obtained using various algorithms: in the above order, αβ-swap algorithm, α-expansion, TRW-S, BP, range swap and range expansion.

in their design, both the methods are faster than previous message passing approaches such as BP and TRW-S. Experiments on synthetic and real data problems demonstrate the effectiveness of our methods compared to several state of the art algorithms.

The two algorithms differ in the *st*-MINCUT problem that they solve at each iteration to move from one labeling to the next. The Range Swap algorithm guarantees that at each iteration the energy of the new labeling obtained by the *st*-MINCUT algorithm is less than or equal to the energy of the previous labeling. However, this monotonic improvement in the energy comes at the price of considering only a subset of the random variables at each iteration. In practice, solving the smaller problem (defined on a subset of random variables) at each iteration makes the Range Swap algorithm computationally efficient. In contrast, the graph construction employed by the Range Expansion algorithm does not guarantee a monotonic improvement in the energy. In other words, the new

54

Algorithm	Energy-1	Time-1(s)	Energy-2	Time-2(s)
αβ-swap	645227	28.86	709120	20.04
α-expansion	634931	9.52	723360	9.78
TRW-S	**634720**	94.86	**651696**	226.07
BP	662108	170.67	2155759	244.71
Range Swap	**634720**	39.75	**651696**	80.40
Range Expansion	**634720**	66.13	**651696**	80.70

(a)

Algorithm	Energy-1	Time-1(s)	Energy-2	Time-2(s)
αβ-swap	1056109	35.00	1198029	52.98
α-expansion	1052860	15.16	1320088	11.95
TRW-S	1053341	142.19	1057371	339.02
BP	1117782	180.65	2443796	368.14
Range Swap	**1052762**	100.49	**1057041**	168.28
Range Expansion	**1052762**	129.30	**1057041**	155.98

(b)

Algorithm	Energy-1	Time-1(s)	Energy-2	Time-2(s)
αβ-swap	3678200	18.48	3707268	20.25
α-expansion	3677950	11.73	3687874	8.79
TRW-S	3677578	131.65	3679563	332.94
BP	3789486	272.06	5180705	331.36
Range Swap	3686844	97.23	**3679552**	141.78
Range Expansion	**3613003**	120.14	**3679552**	191.20

(c)

Table 4: The energy obtained and the time taken by the algorithms used in the stereo reconstruction experiment. Columns 2 and 3 : truncated linear metric. Columns 4 and 5: truncated quadratic semi-metric. **(a)** Tsukuba. **(b)** Venus. **(c)** Teddy. The lowest energy obtained in each case is indicated using bold font.

labeling may have a higher energy than the previous labeling (in which case the new labeling is discarded and the previous labeling is retained). However, the graph construction has the advantage of considering all the random variables at each iteration. The larger search space enables the Range Expansion algorithm to improve the multiplicative bound for the truncated linear metric compared to Boykov et al. (2001) and Gupta and Tardos (2000) and provide the best known bound for the truncated quadratic semi-metric. In practice, the Range Expansion algorithm is computationally more expensive than Range Swap (since the size of the graph is bigger), while providing comparable labelings (with slightly less energy).

Discussion. The speed of both Range Swap and Range Expansion can be further improved by using clever techniques such as those described by Kolmogorov and Shioura (2007) and/or Alahari et al. (2008) for convex and arbitrary unary potentials respectively.

Although we restricted our discussion to truncated convex models for simplicity, our method can easily be extended to handle truncated submodular models by using the graph construction of Schlesinger and Flach (2006) for the submodular part.

The analysis in Section 6.3 shows that, for the truncated linear and truncated quadratic models, the bound achieved by Range Expansion over intervals of any length L is equal to that of rounding the LP relaxation's optimal solution using the same intervals (Chekuri et al., 2005). This equivalence also extends to the Potts model (in which case α-expansion provides the same bound as the LP relaxation with the rounding scheme of Kleinberg and Tardos, 1999) and general metric potentials (in which case the recent method of Kumar and Koller, 2009 provides the same bound as the LP relaxation when using the rounding scheme of Chekuri et al., 2005). This raises the question about the relationship between move making algorithms and the rounding schemes used in convex relaxations. Note that despite recent efforts (Komodakis and Tziritas, 2007) analyzing certain move making algorithms in the context of primal-dual approaches for the LP relaxation, not many results are known about their connection with randomized rounding schemes. Although the discussion in Section 6.3 cannot be trivially generalized to all random fields, it offers a step towards answering this question. We believe that further exploration in this direction would help design efficient move making algorithms for more complex relaxations such as those described in Kumar et al. (2007) and Sontag and Jaakkola (2007).

Acknowledgments

Appendix A. Proof of Theorem 1

Theorem 1: An algorithm that provides a local minimum over smooth labelings achieves a multiplicative bound of 2.

Proof: We denote an optimum labeling by f^* (that is, $Q(f^*, \mathbf{D}; \theta) \leq Q(f, \mathbf{D}; \theta)$ for all labelings f) and a local minimum over smooth labelings by $\hat{f}$. Given f^*, we define a partitioning of the random variables into p subsets using $S_i \subseteq \{0, \cdots, n-1\}$ for $i = 0, 1, \cdots, p-1$ such that $\bigcup_i S_i = \{0, 1, \cdots, n-1\}$ and $S_i \cap S_j = \phi$ (that is, the null set) for all $i \neq j$. In other words, S_i define a disjoint and complete partitioning of the random variables. Furthermore, the subsets S_i are restricted such that f^* is a smooth labeling with respect to S_i. Note that, for any f^*, such a partitioning must exist. This can be seen by observing that the trivial partitioning where each partition consists of only one random variable satisfies the properties described above. In fact, there may be numerous distinct partitionings of the random variables into subsets S_i. Let us take the partitioning that has

the smallest number of subsets, that is p is as small as possible. Note that there are several such minimal partitionings, however the one we shall select does not alter what follows.

Since the subsets S_i define a minimal partitioning, it follows that for any $a \in S_i$ and $b \in S_j$ such that $i \neq j$ and $(a,b) \in \mathcal{E}$, $d(f^*(a) - f^*(b)) > M$. This can easily be proved by contradiction: if there exist $a \in S_i$ and $b \in S_j$ such that $d(f^*(a) - f^*(b)) \leq M$, then we can obtain a smaller partitioning by replacing S_i and S_j by their union. For each subset S_i, we define the following sets

$$
\begin{aligned}
\mathbf{v}(S_i) &= \{v_a | a \in S_i\}, \\
\mathcal{A}(S_i) &= \{(a,b) | (a,b) \in \mathcal{E}, a \in S_i, b \in S_i\}, \\
\mathcal{B}_1(S_i) &= \{(a,b) | (a,b) \in \mathcal{E}, a \in S_i, b \notin S_i\}, \\
\mathcal{B}_2(S_i) &= \{(a,b) | (a,b) \in \mathcal{E}, a \notin S_i, b \in S_i\}, \\
\mathcal{B}(S_i) &= \mathcal{B}_1(S_i) \cup \mathcal{B}_2(S_i).
\end{aligned}
$$

In other words, $\mathbf{v}(S_i)$ contains all the random variables specified by the subset S_i, $\mathcal{A}(S_i)$ contains the set of all edges in the graphical model of the MRF whose endpoints belong to the set $\mathbf{v}(S_i)$ and $\mathcal{B}(S_i)$ contains the set of all edges where only one endpoint belongs to $\mathbf{v}(S_i)$. For each S_i, we also define a labeling f_i such that

$$
f_i(a) = \begin{cases} f^*(a) & \text{if} \qquad a \in S_i, \\ \hat{f}(a) & \text{otherwise.} \end{cases}
$$

Since $\hat{f}$ is a local minimum over smooth labelings, it follows from definition 2 that $Q(\hat{f}, \mathbf{D}; \theta) \leq Q(f_i, \mathbf{D}; \theta)$. By canceling out the common terms, we see that

$$
\begin{aligned}
& \sum_{v_a \in \mathbf{v}(S_i)} \theta_a(\hat{f}(a)) + \sum_{(a,b) \in \mathcal{A}(S_i) \cup \mathcal{B}(S_i)} \theta_{ab}(\hat{f}(a), \hat{f}(b)) \\
\leq \quad & \sum_{v_a \in \mathbf{v}(S_i)} \theta_a(f_i(a)) + \sum_{(a,b) \in \mathcal{A}(S_i) \cup \mathcal{B}(S_i)} \theta_{ab}(f_i(a), f_i(b)) \\
\leq \quad & \sum_{v_a \in \mathbf{v}(S_i)} \theta_a(f^*(a)) + \sum_{(a,b) \in \mathcal{A}(S_i) \cup \mathcal{B}(S_i)} \theta_{ab}(f^*(a), f^*(b)).
\end{aligned}
$$

The last expression holds true because: (i) $\theta_a(f_i(a)) = \theta_a(f^*(a))$ for all $v_a \in \mathbf{v}(S_i)$; (ii) $\theta_{ab}(f_i(a)), f_i(b)) = \theta_{ab}(f^*(a), f^*(b))$ for all $(a,b) \in \mathcal{A}(S_i)$; and (iii) $\theta_{ab}(f^*(a), f^*(b)) = w_{ab}M$ for all $(a,b) \in \mathcal{B}(S_i)$ (since $d(f^*(a) - f^*(b)) > M$). Summing the above inequality over all $i = 0, \cdots, p-1$ and using the fact that the LHS is at least $Q(\hat{f}, \mathbf{D}; \theta)$ we obtain

$$
Q(\hat{f}, \mathbf{D}; \theta) \leq \sum_{v_a \in \mathbf{v}} \theta_a(f^*(a)) + 2 \sum_{(a,b) \in \mathcal{E}} \theta_{ab}(f^*(a), f^*(b)).
$$

The factor 2 in the above inequality appears because the pairwise potential for each $(a,b) \notin \bigcup_i \mathcal{A}(S_i)$ will be counted twice (because it belongs to both $\mathcal{B}_1(S_i)$ and $\mathcal{B}_2(S_j)$ for some i and j). This proves the theorem. ∎

Appendix B. Proof of Lemma 1

Lemma 1: For the capacities defined in Equations (4) and (5), the cost of the *st*-cut which includes the edges (a_k, a_{k+1}) and $(b_{k'}, b_{k'+1})$ (that is, v_a and v_b take labels l_k and $l_{k'}$ respectively) is given by $w_{ab}d(k-k') + \kappa_{ab}$, where the constant $\kappa_{ab} = w_{ab}d(L)$.

Proof: We start by observing that due to the presence of the infinite capacity edges representing unary potentials, the *st*-cut will consist of only the following edges:

$$(a_k, a_{k+1}) \cup (b_{k'}, b_{k'+1}) \cup \{(a_{i'}, b_{j'}), i_m + 1 \le i' \le k, k' + 1 \le j' \le j_m\}$$
$$\cup \{(a_{i'}, b_{j'}), k + 1 \le i' \le k, i_m + 1 \le j' \le k'\}.$$

Using Equations (4) and (5) to sum the capacities of the above edges, we obtain the following expression:

$$\frac{w_{ab}}{2} \left(d(L - k + i_m) + d(k - i_m) \right) + \frac{w_{ab}}{2} \left(d(L - k' + i_m) + d(k' - i_m) \right)$$

$$+ \sum_{i'=i_m+1}^{k} \sum_{j'=k'+1}^{j_m} \frac{w_{ab}}{2} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$

$$+ \sum_{i'=k+1}^{j_m} \sum_{j'=i_m+1}^{k'} \frac{w_{ab}}{2} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right). \tag{13}$$

In order to simplify this expression, consider

$$\sum_{j'=k'+1}^{j_m} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$

$$= \quad d(i' - k') - 2d(i' - k' - 1) + d(i' - k' - 2)$$
$$+ d(i' - k' - 1) - 2d(i' - k' - 2) + d(i' - k' - 3)$$
$$\vdots$$
$$+ d(i' - j_m + 2) - 2d(i' - j_m + 1) + d(i' - j_m)$$
$$+ d(i' - j_m + 1) - 2d(i' - j_m) + d(i' - j_m - 1)$$
$$= \quad d(i' - k') - d(i' - k' - 1) - d(i' - j_m) + d(i' - j_m + 1). \tag{14}$$

Hence, it follows that

$$\sum_{i'=i_m+1}^{k} \sum_{j'=k'+1}^{j_m} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$

$$= \quad d(i_m + 1 - k') - d(i_m - k') - d(i_m - j_m + 1) + d(i_m - j_m)$$
$$+ d(i_m + 2 - k') - d(i_m + 1 - k') - d(i_m - j_m + 2) + d(i_m - j_m + 1)$$
$$\vdots$$
$$+ d(k - k' - 1) - d(k - k' - 2) - d(k - j_m - 1) + d(k - j_m - 2)$$
$$+ d(k - k') - d(k - k' - 1) - d(k - j_m) + d(k - j_m - 1)$$
$$= \quad d(k - k') - d(j_m - k) - d(k' - i_m) + d(j_m - i_m)$$
$$= \quad d(k - k') - d(L - k + i_m) - d(i_m - k') + d(L), \tag{15}$$

where the last expression holds because $L = j_m - i_m$. Note that we also use the fact that $d(x) = d(-x)$. Similarly, it can be shown that

$$\sum_{i'=k+1}^{j_m} \sum_{j'=i_m+1}^{k'} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$

$$= \quad d(k - k') - d(L - k' + i_m) - d(i_m - k) + d(L). \tag{16}$$

58

Substituting Equations (15) and (16) into expression (13), we obtain the cost of the st-cut as

$$\frac{w_{ab}}{2}\left(d(L-k+i_m)+d(k-i_m)\right)+\frac{w_{ab}}{2}\left(d(L-k'+i_m)+d(k'-i_m)\right)$$
$$+\frac{w_{ab}}{2}\left(d(k-k')-d(L-k+i_m)-d(i_m-k')+d(L)\right)$$
$$+\frac{w_{ab}}{2}\left(d(k-k')-d(L-k'+i_m)-d(i_m-k)+d(L)\right)$$
$$=\quad w_{ab}d(k-k')+\kappa_{ab}.$$

This proves that the capacities in Equations (4) and (5) model convex pairwise potentials exactly up to an additive constant. ∎
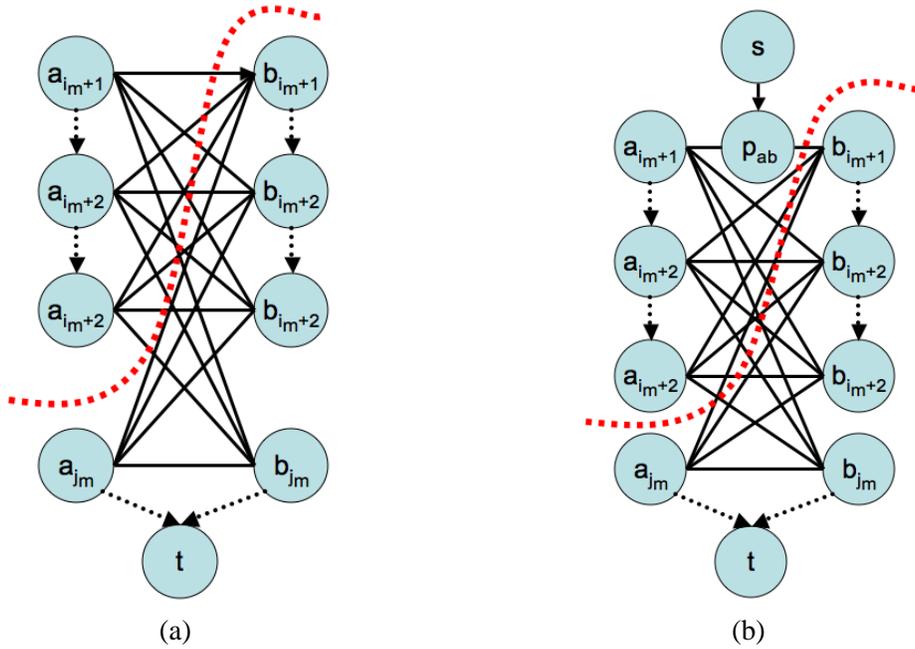

Figure 8: The st-cut (the dashed curve between the two sets of nodes $\{a_{i_m+1},\cdots,a_{j_m}\}$ and $\{b_{i_m+1},\cdots,b_{j_m}\}$; shown in red if viewed in color) that assigns $f(a)\in I_m$ and $f(b)\notin I_m$. Undirected edges represents directed edges in both directions (with the same capacity). **(a)** $f_m(a)\in I_m$ and $f_m(b)\notin I_m$. In this case, we introduce a directed edge from a_{i_m+1} to b_{i_m+1} that is included in the st-cut.**(b)** $f_m(a)\notin I_m$ and $f_m(b)\notin I_m$. In this case, we introduce an auxiliary variable p_{ab} that belongs to the source set in the st-cut.

Appendix C. Proof of Lemma 2

Lemma 2: For the graph described in § 6.1, property 9 holds true.
Proof: We will show the proof for $f(a)\in I_m$ and $f(b)=f_m(b)\notin I_m$. The proof for $f(a)=f_m(a)\notin I_m$ and $f(b)\in I_m$ can be obtained from the following arguments trivially.

There are two possible cases to be considered: (i) $f_m(a) \in I_m$; and (ii) $f_m(a) \notin I_m$. In the first case, the edges that specify the st-cut are given by (see Fig. 8(a))

$$(a_{f(a)}, a_{f(a)+1}) \cup \{(a_{i'}, b_{j'}), i_m + 2 \le i' \le f(a), i_m + 1 \le j' \le j_m\}$$
$$\cup \{(a_{i_m+1}, b_{j'}), i_m + 2 \le j' \le j_m\} \cup (a_{i_m+1}, b_{i_m+1}). \tag{17}$$

In the second case, the st-cut is specified by (see Fig. 8(b))

$$(a_{f(a)}, a_{f(a)+1}) \cup \{(a_{i'}, b_{j'}), i_m + 2 \le i' \le f(a), i_m + 1 \le j' \le j_m\}$$
$$\cup \{(a_{i_m+1}, b_{j'}), i_m + 2 \le j' \le j_m\} \cup (p_{ab}, b_{i_m+1}).$$

Note that in this case p_{ab} belongs to the same partition as the source s. This can be shown easily by observing that the cost of the st-cut increases if p_{ab} belongs to the partition containing the sink t (since this would include edges (a_{i_m+1}, p_{ab}) and (s, p_{ab}) in the st-cut). The two cases differ only in that the first includes the edge (a_{i_m+1}, b_{i_m+1}) and the second includes the edge (p_{ab}, b_{i_m+1}). However, the capacity of both these edges is equal to $w_{ab}M + \kappa_{ab}/2$. Hence it follows that the cost of the st-cut in both the cases is the same. Therefore it is sufficient to show that the Lemma holds true for the first case.

The cost of the st-cut for the edges in Equation (17) is given by

$$\frac{w_{ab}}{2} \left(d(L - f(a) + i_m) + d(f(a) - i_m) \right)$$
$$+ \sum_{i'=i_m+2}^{f(a)} \sum_{j'=i_m+1}^{j_m} \frac{w_{ab}}{2} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$
$$+ \sum_{j'=i_m+2}^{j_m} \frac{w_{ab}}{2} \left(d(i_m - j' + 2) - 2d(i_m - j' + 1) + d(i_m - j') \right)$$
$$+ w_{ab}M + \frac{\kappa_{ab}}{2}. \tag{18}$$

In order to simplify the above expression, we begin by observing that

$$\sum_{j'=i_m+1}^{j_m} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$
$$= \quad d(i' - i_m) - d(i' - i_m - 1) - d(i' - j_m) + d(i' - j_m - 1).$$

The above equation is obtained by substituting $k' = i_m$ in Equation (14). It follows that

$$\sum_{i'=i_m+2}^{f(a)} \sum_{j'=i_m+1}^{j_m} \frac{w_{ab}}{2} \left(d(i' - j' + 1) - 2d(i' - j') + d(i' - j' - 1) \right)$$
$$= \quad d(2) - d(1) - d(i_m - j_m + 2) + d(i_m - j_m + 1)$$
$$+ d(3) - d(2) - d(i_m - j_m + 3) + d(i_m - j_m + 2)$$
$$\vdots$$
$$+ d(f(a) - i_m - 1) - d(f(a) - i_m - 2) - d(f(a) - j_m - 1) + d(f(a) - j_m - 2)$$
$$+ d(f(a) - i_m) - d(f(a) - i_m - 1) - d(f(a) - j_m) + d(f(a) - j_m - 1)$$
$$= \quad d(f(a) - i_m) - d(j_m - f(a)) - d(1) + d(j_m - i_m - 1)$$
$$= \quad d(f(a) - i_m) - d(L - f(a) + i_m) - d(1) + d(L - 1), \tag{19}$$

where the last expression is obtained using $L = j_m - i_m$. Once again, we use the fact that $d(x) = d(-x)$. Similarly, by substituting $k' = i_m + 1$ in Equation (14), we get

$$
\begin{aligned}
& \sum_{j'=i_m+2}^{j_m} \frac{w_{ab}}{2} \left(d(i_m - j' + 2) - 2d(i_m - j' + 1) + d(i_m - j') \right) \\
=\ & d(0) - d(1) - d(j_m - i_m - 1) + d(j_m - i_m) \\
=\ & d(0) - d(1) - d(L - 1) + d(L).
\end{aligned}
\tag{20}
$$

By simplifying expression (18) using Equations (19) and (20), the cost of the st-cut is given by

$$
\begin{aligned}
& \frac{w_{ab}}{2} \left(d(L - f(a) + i_m) + d(f(a) - i_m) \right) \\
& + \frac{w_{ab}}{2} \left(d(f(a) - i_m) - d(L - f(a) + i_m) - d(1) + d(L - 1) \right) \\
& + \frac{w_{ab}}{2} \left(d(0) - d(1) - d(L - 1) + d(L) \right) \\
& + w_{ab}M + \frac{\kappa_{ab}}{2} \\
=\ & w_{ab}d(f(a) - (i_m + 1)) + w_{ab}\hat{d}(f(a) - (i_m + 1)) + w_{ab}M + \kappa_{ab},
\end{aligned}
$$

where the last expression is obtained using the definition of $\hat{d}(\cdot)$ in Equation (10) and the fact that $\kappa_{ab} = w_{ab}d(L)$. This proves the Lemma. ∎

Appendix D. Proof of Lemma 3

Lemma 3: At an iteration of our algorithm, given the current labeling f_m and an interval $I_m = [i_m + 1, j_m]$, the new labeling f_{m+1} obtained by solving the st-MINCUT problem reduces the energy by at least the following:

$$
\begin{aligned}
& \sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f_m(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m) \cup \mathcal{B}(f^*, I_m)} \theta_{ab}(f_m(a), f_m(b)) \\
& - \left(\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f^*(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right).
\end{aligned}
$$

Proof: From the arguments in § 6.2, it is clear that the energy of the new labeling f_{m+1} is bounded from above by the cost of the st-MINCUT. The cost of the st-MINCUT itself is bounded from above by the cost of any other st-cut in the graph $\mathcal{G}_m$. Consider one such st-cut that results in the following labeling:

$$
f(a) = \begin{cases} f^*(a) & \text{if} \qquad v_a \in \mathbf{v}(f^*, I_m) \\ f_m(a) & \text{otherwise}. \end{cases}
$$

We will now derive the cost of this st-cut using the properties in § 6.2. We consider the following six cases:

- For random variables $v_a \notin \mathbf{v}(f^*, I_m)$ it follows from Property 5 that the cost of the st-cut will include the unary potentials associated with such variables exactly, that is,

$$
\sum_{v_a \notin \mathbf{v}(f^*, I_m)} \theta_a(f_m(a)).
\tag{21}
$$

- For neighboring random variables $(a,b) \notin \mathcal{A}(f^*, I_m) \bigcup \mathcal{B}(f^*, I_m)$ it follows from Property 6 that the cost of the st-cut will include the pairwise potentials associated with such neighboring variables exactly up to a constant κ_{ab}, that is,

$$\sum_{(a,b) \notin \mathcal{A}(f^*, I_m) \bigcup \mathcal{B}(f^*, I_m)} (\theta_{ab}(f_m(a), f_m(b)) + \kappa_{ab}) . \tag{22}$$

- For random variables $v_a \in \mathbf{v}(f^*, I_m)$, it follows from Property 5 that the cost of the st-cut will include the unary potentials associated with such variables exactly, that is,

$$\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f^*(a)) . \tag{23}$$

- For neighboring random variables $(a,b) \in \mathcal{A}(f^*, I_m)$ it follows from Properties 7 and 8 that the cost of the st-cut will include the following:

$$\sum_{(a,b) \in \mathcal{A}(f^*, I_m)} (e_{ab}^m + \kappa_{ab}) . \tag{24}$$

- For neighboring random variables $(a,b) \in \mathcal{B}_1(f^*, I_m)$ it follows from Property 9 that the cost of the st-cut will include the following:

$$\sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} (e_a^m + \kappa_{ab}) . \tag{25}$$

- For neighboring random variables $(a,b) \in \mathcal{B}_2(f^*, I_m)$ it follows from Property 9 that the cost of the st-cut will include the following:

$$\sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} (e_b^m + \kappa_{ab}) . \tag{26}$$

The energy of f (that is, $Q(f, \mathbf{D}; \theta)$), and hence $Q(f_{m+1}, \mathbf{D}; \theta)$, is less than or equal to the sum of terms (21)-(26) minus $\sum_{(a,b) \in \mathcal{E}} \kappa_{ab}$. It follows that the difference between the energy of the current labeling f_m and the new labeling f_{m+1}, that is, $Q(f_m, \mathbf{D}; \theta) - Q(f_{m+1}, \mathbf{D}; \theta)$, is at least

$$\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f_m(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m) \bigcup \mathcal{B}(f^*, I_m)} \theta_{ab}(f_m(a), f_m(b))$$

$$- \left(\sum_{v_a \in \mathbf{v}(f^*, I_m)} \theta_a(f^*(a)) + \sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right) .$$

This proves the Lemma. ∎

Appendix E. Proof of Lemma 4

Lemma 4: When $d(\cdot)$ is linear, that is, $d(x) = |x|$, the following inequality holds true:

$$\frac{1}{L} \sum_r \sum_{I_m \in I_r} \left(\sum_{(a,b) \in \mathcal{A}(f^*, I_m)} e_{ab}^m + \sum_{(a,b) \in \mathcal{B}_1(f^*, I_m)} e_a^m + \sum_{(a,b) \in \mathcal{B}_2(f^*, I_m)} e_b^m \right)$$

$$\leq \left(2 + \max \left\{ \frac{2M}{L}, \frac{L}{M} \right\} \right) \sum_{(a,b) \in \mathcal{E}} \theta_{ab}(f^*(a)f^*(b)) . \tag{27}$$

Proof: The following is a slight modification of the proof of Lemma 4.5 of Chekuri et al. (2005) and is presented here for the sake of completeness. Since we are dealing with the truncated linear metric, the terms e_{ab}^m, e_a^m and e_b^m can be simplified as

$$e_{ab}^m = w_{ab}|f^*(a) - f^*(b)|, e_a^m = w_{ab}(f^*(a) - i_m - 1 + M), e_b^m = w_{ab}(f^*(b) - i_m - 1 + M).$$

We begin by observing that the LHS of inequality (27) can be rewritten as

$$\frac{1}{L} \sum_{(a,b) \in \mathcal{E}} \left(\sum_{\mathcal{A}(f^*,I_m) \ni (a,b)} e_{ab}^m + \sum_{\mathcal{B}_1(f^*,I_m) \ni (a,b)} e_a^m + \sum_{\mathcal{B}_2(f^*,I_m) \ni (a,b)} e_b^m \right). \tag{28}$$

In order to prove the Lemma, we consider the following three cases for two neighboring random variables $(a,b) \in \mathcal{E}$.

Case I: $d(f^*(a), f^*(b)) = |f^*(a) - f^*(b)| \leq L$ and hence, $\theta_{ab}(f^*(a), f^*(b)) = w_{ab}M$.

In this case, it is clear that $(a,b) \notin \mathcal{A}(f^*,I_m)$ for all intervals I_m since the length of each interval is L. Furthermore, the conditions for $(a,b) \in \mathcal{B}_1(f^*,I_m)$ and $(a,b) \in \mathcal{B}_2(f^*,I_m)$ are given by

$$(a,b) \in \mathcal{B}_1(f^*,I_m) \iff i_m \in [f^*(a) - L, f^*(a) - 1],$$
$$(a,b) \in \mathcal{B}_2(f^*,I_m) \iff i_m \in [f^*(b) - L, f^*(b) - 1].$$

In order to prove inequality (27), we observe that

$$\sum_{\mathcal{A}(f^*,I_m) \ni (a,b)} e_{ab}^m + \sum_{\mathcal{B}_1(f^*,I_m) \ni (a,b)} e_a^m + \sum_{\mathcal{B}_2(f^*,I_m) \ni (a,b)} e_b^m$$

$$= w_{ab} \left(\sum_{i_m=f^*(a)-L}^{f^*(a)-1} (M + f^*(a) - i_m - 1) + \sum_{i_m=f^*(b)-L}^{f^*(b)-1} (M + f^*(b) - i_m - 1) \right)$$

$$= w_{ab} \left(2LM + \sum_{i_m=f^*(a)-L}^{f^*(a)-1} (f^*(a) - i_m - 1) + \sum_{i_m=f^*(a)-L}^{f^*(a)-1} (f^*(a) - i_m - 1) \right)$$

$$\leq w_{ab} \left(2LM + 2L^2 \right)$$

$$= L \left(2 + \frac{L}{M} \right) \theta_{ab}(f^*(a), f^*(b)), \tag{29}$$

where the last expression is obtained using the fact that $\theta_{ab}(f^*(a), f^*(b)) = w_{ab}M$.

Case II: $M \leq d(f^*(a), f^*(b)) = |f^*(a) - f^*(b)| < L$ and hence, $\theta_{ab}(f^*(a), f^*(b)) = w_{ab}M$.

We will assume, without loss of generality, that $f^*(a) \leq f^*(b)$. In this case, the conditions for $(a,b) \in \mathcal{A}(f^*,I_m)$, $(a,b) \in \mathcal{B}_1(f^*,I_m)$ and $(a,b) \in \mathcal{B}_2(f^*,I_m)$ are given by

$$(a,b) \in \mathcal{A}(f^*,I_m) \iff i_m \in [f^*(b) - L, f^*(a) - 1],$$
$$(a,b) \in \mathcal{B}_1(f^*,I_m) \iff i_m \in [f^*(a) - L, f^*(b) - L - 1],$$
$$(a,b) \in \mathcal{B}_2(f^*,I_m) \iff i_m \in [f^*(a), f^*(b) - 1].$$

Again, in order to prove inequality (27), we observe that

$$\sum_{\mathcal{A}(f^*,I_m)\ni(a,b)} e_{ab}^m + \sum_{\mathcal{B}_1(f^*,I_m)\ni(a,b)} e_a^m + \sum_{\mathcal{B}_2(f^*,I_m)\ni(a,b)} e_b^m$$

$$= w_{ab}\left(\sum_{i_m=f^*(b)-L}^{f^*(a)-1}(f^*(b)-f^*(a)) + \sum_{i_m=f^*(a)-L}^{f^*(b)-L-1}(M+f^*(a)-i_m-1)\right.$$

$$\left.+ \sum_{i_m=f^*(a)}^{f^*(b)-1}(M+f^*(b)-i_m-1)\right)$$

$$\leq w_{ab}\left(2L+2M-(f^*(b)-f^*(a))\right)\left(f^*(b)-f^*(a)\right)$$

$$\leq w_{ab}L(2M+L)$$

$$= L\left(2+\frac{L}{M}\right)\theta_{ab}(f^*(a),f^*(b)),\tag{30}$$

where the last expression is obtained using the fact that $\theta_{ab}(f^*(a),f^*(b)) = w_{ab}M$.

Case III: $d(f^*(a),f^*(b)) = |f^*(a)-f^*(b)| \leq M$ and hence, $\theta_{ab}(f^*(a),f^*(b)) = w_{ab}|f^*(a)-f^*(b)|$.

We will assume, without loss of generality, that $f^*(a) \leq f^*(b)$. Similar to case II, the conditions for $(a,b) \in \mathcal{A}(f^*,I_m)$, $(a,b) \in \mathcal{B}_1(f^*,I_m)$ and $(a,b) \in \mathcal{B}_2(f^*,I_m)$ are given by

$$(a,b) \in \mathcal{A}(f^*,I_m) \Longleftrightarrow i_m \in [f^*(b)-L, f^*(a)-1],$$

$$(a,b) \in \mathcal{B}_1(f^*,I_m) \Longleftrightarrow i_m \in [f^*(a)-L, f^*(b)-L-1],$$

$$(a,b) \in \mathcal{B}_2(f^*,I_m) \Longleftrightarrow i_m \in [f^*(a), f^*(b)-1].$$

Once again, we consider

$$\sum_{\mathcal{A}(f^*,I_m)\ni(a,b)} e_{ab}^m + \sum_{\mathcal{B}_1(f^*,I_m)\ni(a,b)} e_a^m + \sum_{\mathcal{B}_2(f^*,I_m)\ni(a,b)} e_b^m$$

$$= w_{ab}\left(\sum_{i_m=f^*(b)-L}^{f^*(a)-1}(f^*(b)-f^*(a)) + \sum_{i_m=f^*(a)-L}^{f^*(b)-L-1}(M+f^*(a)-i_m-1)\right.$$

$$\left.+ \sum_{i_m=f^*(a)}^{f^*(b)-1}(M+f^*(b)-i_m-1)\right)$$

$$\leq w_{ab}\left(2L+2M-(f^*(b)-f^*(a))\right)\left(f^*(b)-f^*(a)\right)$$

$$\leq w_{ab}(2L+2M)(f^*(b)-f^*(a))$$

$$= L\left(2+\frac{2M}{L}\right)\theta_{ab}(f^*(a),f^*(b)),\tag{31}$$

where the last expression is obtained using the fact that $\theta_{ab}(f^*(a),f^*(b)) = w_{ab}(f^*(b)-f^*(a))$.

Substituting inequalities (29), (30) and (31) in expression (28) and dividing both sides by L for all $(a,b) \in \mathcal{E}$, we obtain inequality (27). This proves the Lemma. ∎

References

K. Alahari, P. Kohli, and P. H. S. Torr. Reduce, reuse & recycle: Efficiently solving multi-label MRFs. In *CVPR*, 2008.

J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B*, 48:259–302, 1986.

S. Birchfield and C. Tomasi. Depth discontinuities by pixel-to-pixel stereo. In *ICCV*, pages 1073–1080, 1998.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

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

Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, 2001.

C. Chekuri, S. Khanna, J. Naor, and L. Zosin. A linear programming formulation and approximation algorithms for the metric labelling problem. *SIAM Journal on Discrete Mathematics*, 18(3):606–635, 2005.

E. Dinic. Algorithm for solution of a problem of maximum flow in networks with power estimation. *Soviet Math. Dokl.*, 11:1277–1280, 1970.

P. Felzenszwalb and D. Huttenlocher. Efficient matching of pictorial structures. In *CVPR*, pages II: 66–73, 2000.

P. Felzenszwalb and D. Huttenlocher. Efficient belief propagation for early vision. In *CVPR*, pages I: 261–268, 2004.

A. Globerson and T. Jaakkola. Fixing max-product: Convergent message passing for MAP LP-relaxations. In *NIPS*, 2007.

A. Goldberg and R. Tarjan. A new approach to the maximum-flow problem. *Journal of ACM*, 35 (4):921–940, 1988.

S. Gould, F. Amat, and D. Koller. Alphabet SOUP: A framework for approximate energy minimization. In *CVPR*, 2009.

A. Gupta and E. Tardos. A constant factor approximation algorithm for a class of classification problems. In *STOC*, 2000.

H. Ishikawa. Exact optimization for Markov random fields with convex priors. *PAMI*, 25(10): 1333–1336, October 2003.

J. Kleinberg and E. Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and Markov random fields. In *STOC*, pages 14–23, 1999.

V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28 (10):1568–1583, 2006.

V. Kolmogorov and A. Shioura. New algorithms for the dual of the convex cost network flow problem with applications to computer vision. Technical report, University College London, 2007.

N. Komodakis and G. Tziritas. Approximate labeling via graph-cuts based on linear programming. *PAMI*, 2007.

N. Komodakis, N. Paragios, and G. Tziritas. MRF optimization via dual decomposition: Message-passing revisited. In *ICCV*, 2007.

A. Koster, C. van Hoesel, and A. Kolen. The partial constraint satisfaction problem: Facets and lifting theorems. *Operations Research Letters*, 23(3-5):89–97, 1998.

M. P. Kumar and D. Koller. MAP estimation of semi-metric MRFs via hierarchical graph cuts. In *UAI*, 2009.

M. P. Kumar and P. H. S. Torr. Improved moves for truncated convex models. In *NIPS*, 2008.

M. P. Kumar, V. Kolmogorov, and P. H. S. Torr. An analysis of convex relaxations for MAP estimation. In *NIPS*, 2007.

J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labelling sequence data. In *ICML*, 2001.

T. Meltzer, C. Yanover, and Y. Weiss. Globally optimal solutions for energy minimization in stereo vision using reweighted belief propagation. In *ICCV*, 2005.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kauffman, 1988.

P. Ravikumar, A. Agarwal, and M. Wainwright. Message-passing for graph-structured linear programs: Proximal projections, convergence and rounding schemes. In *ICML*, 2008.

D. Schlesinger and B. Flach. Transforming an arbitrary minsum problem into a binary one. Technical Report TUD-F106-01, Dresden University of Technology, 2006.

M. Schlesinger. Sintaksicheskiy analiz dvumernykh zritelnikh singnalov v usloviyakh pomekh (syntactic analysis of two-dimensional visual signals in noisy conditions). *Kibernetika*, 4:113–130, 1976.

M. Schlesinger and V. Giginyak. Solution to structural recognition (MAX,+)-problems by their equivalent transformations. Part 1. *Control Systems and Computers*, 1:3–15, 2007a.

M. Schlesinger and V. Giginyak. Solution to structural recognition (MAX,+)-problems by their equivalent transformations. Part 2. *Control Systems and Computers*, 2:3–18, 2007b.

J. Shotton, J. Winn, C. Rother, and A. Criminisi. TextonBoost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*, pages I: 1–15, 2006.

D. Sontag and T. Jaakkola. New outer bounds on the marginal polytope. In *NIPS*, 2007.

R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *PAMI*, 2008.

V. Vazirani. *Approximation Algorithms*. Springer-Verlag, 2001.

O. Veksler. *Efficient Graph-Based Energy Minimization Methods in Computer Vision*. PhD thesis, Cornell University, 1999.

O. Veksler. Graph cut based optimization for MRFs with truncated convex priors. In *CVPR*, 2007.

M. Wainwright, T. Jaakkola, and A. Willsky. MAP estimation via agreement on trees: Message passing and linear programming. *IEEE Trans. on Information Theory*, 51(11):3697–3717, 2005.