Generalized Fast Approximate Energy Minimization via Graph Cuts: Alpha-Expansion Beta-Shrink Moves

Mark Schmidt, Karteek Alahari

▶ **To cite this version:**

Mark Schmidt, Karteek Alahari. Generalized Fast Approximate Energy Minimization via Graph Cuts: Alpha-Expansion Beta-Shrink Moves. UAI 2011 - 27th Conference on Uncertainty in Artificial Intelligence, Jul 2011, Barcelona, Spain. inria-00617524

Generalized Fast Approximate Energy Minimization via Graph Cuts: α-Expansion β-Shrink Moves

Mark Schmidt
INRIA - SIERRA Team*
Laboratoire d'Informatique
École Normale Supérieure
Paris, France

Karteek Alahari
INRIA - WILLOW Team*
Laboratoire d'Informatique
École Normale Supérieure
Paris, France

Abstract

We present α-expansion β-shrink moves, a simple generalization of the widely-used $\alpha\beta$-swap and α-expansion algorithms for approximate energy minimization. We show that in a certain sense, these moves *dominate* both $\alpha\beta$-swap and α-expansion moves, but unlike previous generalizations the new moves require no additional assumptions and are still solvable in polynomial-time. We show promising experimental results with the new moves, which we believe could be used in any context where α-expansions are currently employed.

1   Introduction

We focus on the problem of finding the most probable configuration in a pairwise Markov random field over discrete variables, a fundamental problem in the study of graphical models. This is equivalent to minimizing the sum of a set of unary and pairwise energy functions defined over a set of discrete variables. Problems of this type arise in many applications, but in general it is NP-hard to solve these problems even in the case of binary variables (see Kolmogorov and Zabih, 2002, Theorem 4.2). A classical method for computing an approximate solution to this problem is Besag's iterated conditional mode (ICM) algorithm (Besag, 1986), but due to the local nature of this method it may become stuck in a poor local optimum.

In the special case where the variables are binary and the pairwise energy functions satisfy a submodularity condition, it is possible to solve this problem in polynomial-time (Hammer, 1965; Greig et al., 1989; Kolmogorov and Zabih, 2002). This has motivated the $\alpha\beta$-swap and α-expansion moves proposed by Boykov

et al. (1998, 1999), which we review in §2. These methods find an approximate solution to a non-binary problem satisfying a submodularity condition over pairs of states (or triplets of states in the case of α-expansions) by solving a sequence of binary submodular problems. The experiments of Szeliski et al. (2008) show that these algorithms perform well compared to competing approximate minimization methods at minimizing the energy functions arising from problems in computer vision, such as stereo image matching, creating photo montages, and image restoration. Further, this study found that α-expansions are often faster than competing methods with similar performance for these problems (such as variational message-passing algorithms like tree-reweighted belief propagation). While originally introduced in the context of computer vision, these algorithms and their generalizations have also proved to be effective in other domains, such as protein structure prediction (Gould et al., 2009).

In a sense that we formally define in §3, $\alpha\beta$-swaps and α-expansions *dominate* the classic ICM algorithm. However, although α-expansions often lead to better experimental results than $\alpha\beta$-swaps (Szeliski et al., 2008), under our definition of dominance neither of these more advanced methods dominates the other. In this paper we propose a new type of move, α-expansion β-shrink moves (§4). These move are a simple generalization of both $\alpha\beta$-swap and α-expansions, that *dominates them both*. Although we delay discussion of other generalizations of these moves to the end (§6), we note that unlike previous generalizations the new moves require no additional assumptions beyond those needed to apply α-expansions, and the moves can be computed in polynomial-time by solving a binary submodular problem defined on the original graph structure. Thus, we can use them in place of α-expansions for applications where these moves are currently used. Our experiments on standard test problems from the field of computer vision (§5) show that the new moves can lead to improved performance.

---

*INRIA/ENS/CNRS UMR 8548

2 Approximate Energy Minimization

The problem that we formally address is

$$\max_{x \in \{1,2,\ldots,N\}^p} p(x) \propto \prod_{i \in \mathcal{V}} \phi_i(x_i) \prod_{(i,j) \in \mathcal{A}} \phi_{ij}(x_i, x_j),$$

where $\mathcal{V}$ and $\mathcal{A}$ are the vertices and arcs of a graph $(\mathcal{V}, \mathcal{A})$, while the potentials $\phi_i(x_i)$ and $\phi_{ij}(x_i, x_j)$ map assignments of subsets of the discrete vector x to non-negative values. If we define the real-valued energy functions

$$E_i(x_i) = -\log \phi_i(x_i), \quad E_{ij} = -\log \phi_{ij}(x_i, x_j),$$

then finding the optimal assignment is equivalent to the following energy minimization problem:

$$\min_{x \in \{1,2,\ldots,N\}^p} \sum_{i \in \mathcal{V}} E_i(x_i) + \sum_{(i,j) \in \mathcal{A}} E_{ij}(x_i, x_j), \qquad (1)$$

In general, solving this optimization problem is NP-hard (see Kolmogorov and Zabih, 2002, Theorem 4.2), and a common approach to finding an approximate minimizer is with an iterative descent algorithm. The input to each iteration of these algorithms is a particular configuration of the variables x, and at each iteration an iterative descent method finds a configuration that minimizes the energy among a set $\mathcal{M}(x)$ of possible 'moves', i.e. the next iteration is an element of

$$\arg\min_{y \in \mathcal{M}(x)} \sum_{i \in \mathcal{V}} E_i(y_i) + \sum_{(i,j) \in \mathcal{A}} E_{ij}(y_i, y_j). \qquad (2)$$

We have two conflicting desiderata on the set of moves $\mathcal{M}(x^k)$: we would like this set to be as large as possible, but we would like to able to efficiently find the optimal move.

2.1 Iterated Conditional Mode

In the classic ICM algorithm (Besag, 1986), a node j is selected and we replace x_j with a value that maximizes the conditional probability $p(x_j|x_{-j})$, where x_{-j} are the states of all variables except j. In the framework of energy minimization, this can be viewed as an iterative descent method where the elements y of the set of possible moves have the form

$$y_i \leftarrow \begin{cases} \gamma \in \{1, 2, \ldots, N\} & \text{if } i = j, \\ x_i & \text{otherwise.} \end{cases}$$

That is, the current state of node j can be replaced by any other possible state, a form of coordinate descent. We use $\mathcal{M}_j^I(x)$ to denote the set of all y of this form. With this definition of the move space, the iterative

descent update (2) for the ICM move given x and j simplifies to

$$\arg\min_{y \in \mathcal{M}_j^I(x)} E_j(y_j|x_{-j}), \qquad (3)$$

where we will find it convenient to define the *conditional energy* of a variable i given a set of variables a as

$$\begin{aligned} E_i(x_i|x_a) &= E_i(x_i) \\ &+ \sum_{j|j \in a, (i,j) \in \mathcal{A}} E_{ij}(y_i, x_j) \\ &+ \sum_{j|j \in a, (j,i) \in \mathcal{A}} E_{ij}(x_j, y_i). \end{aligned} \qquad (4)$$

While $\exp(-E_i(x_i|x_{-i})) \propto p(x_i|x_{-i})$, this is a slight abuse of the conditioning notation since for other conditioning sets a it ignores factors that depend on variables besides i and those in a. Clearly, we can efficiently compute the optimal ICM move by simply testing each $E_j(y_j|x_{-j})$.

2.2 $\alpha\beta$-Swaps

We say that a pairwise energy function E_{ij} defined on binary variables is *submodular* if it satisfies the inequality[1]

$$E_{ij}(1,1) + E_{ij}(2,2) \leq E_{ij}(2,1) + E_{ij}(1,2). \qquad (5)$$

This type of pairwise energy prefers the neighbouring variables to take the same state. In the special case of binary variables where all pairwise energies are submodular, the optimal solution to (1) can be computed in polynomial-time as a minimum-cut problem, see (Kolmogorov and Zabih, 2002).[2]

In some non-binary problems we have, for all combinations of states α and β, that the pairwise energies E_{ij} satisfy

$$E_{ij}(\alpha, \alpha) + E_{ij}(\beta, \beta) \leq E_{ij}(\beta, \alpha) + E_{ij}(\alpha, \beta). \qquad (6)$$

That is, when restricted to any two states α and β, the energy function is submodular. Though we can no longer guarantee that we can find the optimal solution in polynomial-time given only this restriction, it

---

[1] Submodularity is normally defined as a property of functions on sets, and its use here is because it is equivalent to submodularity of a function that takes the set of variables labeled 2 and returns the corresponding E_{ij} (Kolmogorov and Zabih, 2002, §7).

[2] There also exist several other notable cases where it possible to compute the solution in polynomial-time, such as the case where the pairwise energies are convex (Ishikawa, 2003), or in the case of general energies where the graph structure has low treewidth (Koller and Friedman, 2009, §13) or is outer-planar (Schraudolph and Kamenetsky, 2009). However, these are not our focus and a full discussion of this extensive literature is outside the scope of the current work.

does allow us to take advantage of the ability to efficiently solve binary submodular problems in order to implement a more powerful descent move.

In particular, given two states α and β, the set of moves $\mathcal{M}^S_{\alpha\beta}(x)$ associated with the $\alpha\beta$-swap move introduced by Boykov et al. (1998) are of the form:

$$y_i \leftarrow \begin{cases} \alpha \text{ or } \beta & \text{if } x_i = \alpha \text{ or } x_i = \beta, \\ x_i & \text{otherwise.} \end{cases}$$

That is, the move can simultaneously change any combination of nodes labeled α to β, and any combination of nodes labeled β to α. In this case, the iterative descent update is a solution of the problem

$$\operatorname*{argmin}_{y \in \mathcal{M}^S_{\alpha\beta}(x)} \sum_{i \in \mathcal{V}|x_i \in \{\alpha,\beta\}} E_i(y_i|x_{-\alpha\beta}) \\ + \sum_{(i,j) \in \mathcal{A}|x_i,x_j \in \{\alpha,\beta\}} E_{ij}(y_i, y_j),$$

where we again make use of our definition of the conditional energy (4) and where we have used $x_{-\alpha\beta}$ to reference the states of the variables not labeled α or β. This is a binary problem over the subgraph induced by the nodes labeled α or β, and under condition (6) this update can be computed in polynomial-time because all edges in the induced subgraph are submodular.

2.3 α-Expansions

A closely-related set of moves later proposed by Boykov et al. (1999) are α-expansions. Here, we choose a state α and we can use α to replace the current state of any variable. Thus, this set of moves $\mathcal{M}^E_\alpha(x)$ is of the form

$$y_i \leftarrow \begin{cases} \alpha & \text{if } x_i = \alpha, \\ \alpha \text{ or } x_i & \text{otherwise.} \end{cases}$$

We can write the optimal α-expansion as the solution to the problem

$$\operatorname*{argmin}_{y \in \mathcal{M}^E_\alpha(x)} \sum_{i \in \mathcal{V}|x_i \neq \alpha} E_i(y_i|x_\alpha) \\ + \sum_{(i,j) \in A|x_i \neq \alpha, x_j \neq \alpha} E_{ij}(y_i, y_j),$$

where we use x_α to reference variables labeled α. This is again a binary problem, this time on the subgraph induced by those nodes not labeled α. However, condition (6) is no longer sufficient to guarantee that the edges in the induced subgraph are submodular since each pairwise term involves the three states α, x_i, and x_j (which may all be different). Nevertheless, it is sufficient to ensure that

$$E_{ij}(\alpha, \alpha) + E_{ij}(\gamma_1, \gamma_2) \leq E_{ij}(\gamma_1, \alpha) + E_{ij}(\alpha, \gamma_2), \quad (7)$$

for all combinations of states α, γ_1, and γ_2. This is a stronger condition than (6), which corresponds to the special case where $\gamma_1 = \gamma_2$. If $E_{ij}(\alpha, \alpha) = 0$ for all α, this is the triangle inequality.

3 Local Dominance of Iterative Algorithms

Some authors analyze the approximation accuracy of local optima found by iterative descent methods such as α-expansions (Veksler, 1999, §4.3.4), but the guarantees on the approximation accuracy tend to be very conservative since these global bounds must hold over all possible optima. Further, they often do not provide guidance in choosing between iterative descent methods. For example, a small modification to the analysis of (see Veksler, 1999, §4.3.4) gives us an identical global bound for the ICM algorithm.

Instead of focusing on such a global analysis, we instead focus on a simple local analysis of the methods. Specifically, we are interested in the improvement in the objective function that can be obtained in a single iteration with different types of moves, given the same input to the iteration. From this viewpoint, we say that a move set $\mathcal{M}^A(x)$ *dominates* a move set $\mathcal{M}^B(x)$ if:

1. For any energy function $E(x)$ and any input configuration x, optimizing over $\mathcal{M}^A(x)$ never gives a higher energy than optimizing over $\mathcal{M}^B(\text{x})$:

$$\forall_{E,x}, \min_{y \in \mathcal{M}^A(x)} E(x) \leq \min_{y \in \mathcal{M}^B(x)} E(x). \quad (8)$$

2. There exists energy functions $E(x)$ and input configurations x where optimizing over $\mathcal{M}^A(x)$ gives a strictly lower energy than optimizing over $\mathcal{M}^B(x)$:

$$\exists_{E,x}, \min_{y \in \mathcal{M}^A(x)} E(x) < \min_{y \in \mathcal{M}^B(x)} E(x). \quad (9)$$

This is an intuitive notion of the relative local improvement that is made by different moves, but it can also be used as a measure of the relative 'strengths' of local optima found by different types of moves; if the move set $\mathcal{M}^A(x)$ dominates the move set $\mathcal{M}^B(x)$, then no move in the set $\mathcal{M}^B(x)$ will ever be able to improve on a local minimum with respect to move set $\mathcal{M}^A(x)$, but the optimal move in move set $\mathcal{M}^A(x)$ may be able to escape from a local minima with respect to move set $\mathcal{M}^B(x)$.

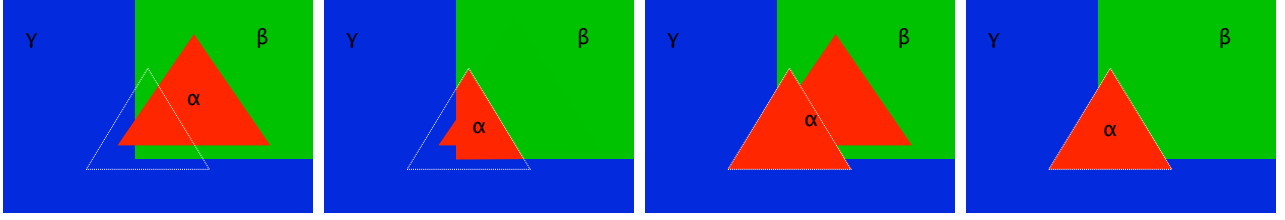To establish dominance relations between various search moves, it will be convenient to define versions

Figure 1: From left to right: Initial labeling, labeling after $\alpha\beta$-swap, labeling after α-expansion, labeling after α-expansion β-shrink. The optimal labeling of the α pixels is outlined by a white triangle, and is achieved from the initial labeling by one α-expansion β-shrink move.

of the moves that do not depend on any parameters except x. Thus, we make the following definitions:

$$\mathcal{M}^I(x) = \cup_j \mathcal{M}_j^I(x),$$
$$\mathcal{M}^S(x) = \cup_{\alpha,\beta} \mathcal{M}_{\alpha\beta}^S(x),$$
$$\mathcal{M}^E(x) = \cup_\alpha \mathcal{M}_\alpha^E(x).$$

That is, these expanded move spaces simply search over all possible values of the parameter(s). Under these definitions, it is straightforward to establish the following relationships between the methods:

Proposition 1. *The $\alpha\beta$-swap move set $\mathcal{M}^S(x)$ dominates the ICM move set $\mathcal{M}^I(x)$. The α-expansion move set $\mathcal{M}^E(x)$ dominates the ICM move set $\mathcal{M}^I(x)$.*

Proof. Assume an arbitrary energy function E and configuration x, and let y be the optimal move in $\mathcal{M}^I(x)$, which must be an element of $\mathcal{M}_j^I(x)$ for some j. Then the optimal ICM move is in $\mathcal{M}_{\alpha\beta}^S(x)$ with $\alpha = x_j$ and $\beta = y_j$, which establishes (8). To establish (9), it is sufficient to construct an energy function E along with a configuration x where the optimal solution involves changing two elements of x. For example, take a two variable problem with $x = (1,1)$ where the optimal solution is $(2,2)$. Establishing that α-expansions dominate ICM can be done similarly. $\square$

Proposition 2. *The α-expansion move set $\mathcal{M}^E(x)$ does not dominate the $\alpha\beta$-swap move set $\mathcal{M}^S(x)$. The $\alpha\beta$-swap move set $\mathcal{M}^S(x)$ does not dominate the α-expansion move set $\mathcal{M}^E(x)$.*

Proof. To establish that α-expansions do not dominate $\alpha\beta$-swaps, it is sufficient to construct an energy function E and configuration x where the optimal solution is obtained by a single $\alpha\beta$-swap but can not be obtained by a single α-expansion. For example, take a two-variable problem with $x = (1,2)$ where the optimal solution is $(2,1)$. We can similarly show that $\alpha\beta$-swaps do not dominate α-expansions by taking a three variable problem with $x = (1,2,3)$ where the optimal solution is $(1,1,1)$. $\square$

Thus, based on the definition of dominance discussed in this section, we should prefer both $\alpha\beta$-swaps and α-expansions to ICM. However, local dominance does not guide us in selecting between these two more advanced methods.

4 α-Expansion β-Shrink Moves

Towards the goal of developing a method that dominates both these methods, consider a generalization of both $\alpha\beta$-swaps and α-expansions where the set of moves $\mathcal{M}_{\alpha\beta}^G(x)$ are of the form:

$$y_i \leftarrow \begin{cases} \alpha \text{ or } \beta & \text{if } x_i = \alpha \\ \alpha \text{ or } x_i & \text{otherwise} \end{cases}$$

That is, we allow any node not currently labeled α to take the value α, but in addition we allow the nodes currently labeled α to take the value β (α expands everywhere, but is shrunk by β). We call these α-expansion β-shrink moves, and Figure 2.3 illustrates an instance of the move.

Under the definition of the previous section, we have the following result for an analogously defined $\mathcal{M}^G(x)$:

Proposition 3. *The α-expansion β-shrink move set $\mathcal{M}^G(x)$ dominates the $\alpha\beta$-swap move set $\mathcal{M}^S(x)$. The α-expansion β-shrink move set $\mathcal{M}^G(x)$ dominates the α-expansion move set $\mathcal{M}^E(x)$.*

Proof. To establish (8), we note that $\mathcal{M}^G(x)$ includes all $\alpha\beta$-swaps and α-expansions as special cases. Further, it also includes moves that are not instances of either of these moves (i.e. cases where α expands into more than one state, but β shrinks into α). Thus, to establish (9) we can simply take a three variable problem with $x = (1,2,3)$ where the optimal solution is $(2,1,1)$. $\square$

Note that the argument above can also be used to show that the new move set $\mathcal{M}^G(x)$ also dominates the move set $\mathcal{M}^S \cup \mathcal{M}^E$, the union of $\alpha\beta$-swap and α-expansion moves.

We can write the optimal α-expansion β-shrink move as a solution to the problem

$$\operatorname*{argmin}_{y \in \mathcal{M}_{\alpha\beta}^G(x)} \sum_{i \in \mathcal{V}} E_i(y_i) + \sum_{(i,j) \in \mathcal{A}} E_{ij}(y_i, y_j), \qquad (10)$$

Note that unlike the previous move sets we have discussed where the move is computed by solving a problem involving conditional energies over an induced subgraph, this move involves solving a problem with the original unary energies on the original graph structure. In some scenarios, these properties might make the generalized move simpler to implement than $\alpha\beta$-swaps or α-expansions. Further, if the original graph has a special structure, it allows the use of specialized codes for solving binary minimum-cut problems with this structure. For example, Delong and Boykov (2008) propose a method for the special case of problems with very large grid structures. The simple form of the subproblem may also simplify the implementation of *dynamic* graph cuts (Kohli and Torr, 2007), where the similarity between subproblems when using the same value of α (and β) is used to substantially speed up the computation.

There are many possible moves we could define that would dominate one or more of the methods discussed in §2, and we return to this topic in §6. However, an appealing property of α-expansion β-shrink moves is the following:

Proposition 4. *If for each edge E_{ij} condition* (7) *holds for all γ_1 and γ_2, then all edge energies E_{ij} in problem* (10) *are submodular.*

Proof. It is sufficient to show this for an arbitrary E_{ij} and any possible assignment to x_i and x_j. If $x_i \neq \alpha$ and $x_j \neq \alpha$, then we require $E_{ij}(\alpha, \alpha) + E_{ij}(x_i, x_j) \leq E_{ij}(x_i, \alpha) + E_{ij}(\alpha, x_j)$ as with α-expansion moves, which is (7) with $\gamma_1 = x_i$ and $\gamma_2 = x_j$. If $x_i = \alpha$ and $x_j = \alpha$, then we require $E_{ij}(\alpha, \alpha) + E_{ij}(\beta, \beta) \leq E_{ij}(\beta, \alpha) + E_{ij}(\alpha, \beta)$ as with $\alpha\beta$-swap moves, which is (7) with $\gamma_1 = \beta$ and $\gamma_2 = \beta$. If $x_i = \alpha$ and $x_j \neq \alpha$, then we require $E_{ij}(\alpha, \alpha) + E_{ij}(\beta, x_j) \leq E_{ij}(\beta, \alpha) + E_{ij}(\alpha, x_j)$, which is (7) with $\gamma_1 = \beta$ and $\gamma_2 = x_j$. The remaining possibility is similar. $\square$

This implies that an optimal α-expansion β-shrink move can be computed by solving a minimum-cut problem, using (for example) the construction of Kolmogorov and Zabih (2002). Thus, an optimal α-expansion β-shrink move can be computed in polynomial time under the same condition required to use α-expansions. Further, computing an optimal α-expansion move and computing an optimal α-expansion β-shrink move have the same worst-case time-complexity. This would indicate that the new moves could be used in place of α-expansions in any of the many applications where these moves are currently used.

4.1 Problems with Many States

In the computer vision problems that originally motivated the moves based on minimum-cuts, the number of states N may be non-trivial, since the states may represent a discretization of a continuous value. For example, Szeliski et al. (2008) evaluate the performance of these moves on problems with 256 states, each state representing an intensity level in an image. Thus, in addition to their better empirical performance, α-expansions may be preferred over $\alpha\beta$-swaps simply because the number of possible values of α is N, while the number of (non-exchangeable) combinations of α and β is $N(N-1)/2$.

This quadratic scaling in terms of N would also seem to be a problem for α-expansion β-shrink moves. However, if we consider *any* strategy for choosing a value of β given α, then we still obtain an algorithm that dominates α-expansions (assuming we do not always choose $\alpha = \beta$, corresponding to the special case of α-expansions). Thus, the new moves can be modified to have the same scaling with N as α-expansions. That is, if N is very large we can consider variants where β is function of α so that we only consider N moves instead of N^2, and these variants would still dominate α-expansions. We empirically evaluate three possible strategies for selecting β given α in §5.

4.2 Truncation for Non-Submodular Potentials

In many problems condition (7) is not satisfied. In these cases, a widely-used approach is to modify the potentials to be submodular, in such a way that an optimal move with the modified energy is guaranteed to not increase the original energy (Rother et al., 2005). In the case of α-expansions, one way to construct such a modified energy is by replacing each $E_{ij}(x_i, x_j)$ with

$$\bar{E}_{ij}(x_i, x_j) = \min\{E_{ij}(x_i, x_j),$$
$$E_{ij}(\alpha, x_j) + E_{ij}(x_i, \alpha) - E_{ij}(\alpha, \alpha)\}.$$

Condition (7) holds with this modified energy by construction. Further, the optimal α-expansion with this modified energy does not increase the original energy, since the modified energy simply decreases the energy of the current assignment (x_i, x_j). As discussed in (Rother et al., 2005), we can alternately increase $E_{ij}(x_i, \alpha)$ or $E_{ij}(\alpha, x_j)$ to make condition (7) satisfied while maintaining the descent property of the moves.

We can define a modified energy function with similar properties in the case of α-expansion β-shrink

moves, though maintaining the descent property requires a slightly more complicated construction. Although there are many possible constructions, we describe one here. If $x_i \neq \alpha$ and $x_j \neq \alpha$, then as before we take:

$$\bar{E}_{ij}(x_i, x_j) = \min\{E_{ij}(x_i, x_j),$$
$$E_{ij}(\alpha, x_j) + E_{ij}(x_i, \alpha) - E_{ij}(\alpha, \alpha)\}.$$

If $x_i = \alpha$ and $x_j = \alpha$, then we take:

$$\bar{E}_{ij}(\alpha, \alpha) = \min\{E_{ij}(\alpha, \alpha),$$
$$E_{ij}(\alpha, \beta) + E_{ij}(\beta, \alpha) - E_{ij}(\beta, \beta)\}.$$

If $x_i \neq \alpha$ and $x_j = \alpha$, then we take:

$$\bar{E}_{ij}(\alpha, \beta) = \max\{E_{ij}(\alpha, \beta),$$
$$E_{ij}(\alpha, \alpha) + E_{ij}(x_i, \beta) - E_{ij}(x_i, \alpha)\}.$$

Finally, if $x_i = \alpha$ with $x_j \neq \alpha$ we take:

$$\bar{E}_{ij}(\beta, \alpha) = \max\{E_{ij}(\beta, \alpha),$$
$$E_{ij}(\alpha, \alpha) + E_{ij}(\beta, x_j) - E_{ij}(\alpha, x_j)\}.$$

The other terms in the energy function are unchanged. If (7) is already satisfied, then the modified energy under this construction is identical to the original energy. This construction maintains the appealing property that any move that does not increase the modified energy will not increase the original energy.

5 Experiments

To empirically evaluate the performance of the new moves, we performed several experiments on the non-binary data sets examined by Szeliski et al. (2008). These data sets are summarized in Table 1, and we extracted the terms in the energy functions from the code available online.[3]

Our experiments compared the following methods:

- $\alpha\beta$-Swap: performing $\alpha\beta$-swap moves in the order $\beta = 1, 2, \ldots, N$ in an outer loop and $\alpha = \beta+1, \beta+2, \ldots, N$ in an inner loop.

- α-Expansion: performing α-expansion moves in the order $\alpha = 1, 2, \ldots, N$.

- Random β: performing α-expansion β-shrink moves in the order $\alpha = 1, 2, \ldots, N$ with β selected randomly among $\{1, 2, \ldots, N\}$.

- $\beta = \alpha - 1$: performing α-expansion β-shrink moves in the order $\alpha = 1, 2, \ldots, N$ with β set to $\max\{1, \alpha - 1\}$.

---

[3]http://vision.middlebury.edu/MRF/

Table 1: Data sets from Szeliski et al. (2008) used in the experiments.

Name	Task	Nodes	Edges	States
Family	Montage	425632	849946	5
Pano	Montage	514080	1026609	7
Tsukuba	Stereo	110592	220512	16
Venus	Stereo	166222	331627	20
Teddy	Stereo	168750	336675	60
Penguin	Restoration	21838	43375	256
House	Restoration	65536	130560	256

- $\beta = \alpha + 1$: performing α-expansion β-shrink moves in the order $\alpha = 1, 2, \ldots, N$ with β set to $\min\{N, \alpha + 1\}$.

- All β: performing α-expansion β-shrink moves in the order $\beta = 1, 2, \ldots, N$ in an outer loop and $\alpha = 1, 2, \ldots, N$ in an inner loop.

Note that the iterations of the first and last method are *much* more expensive for large N because they consider $\mathcal{O}(N^2)$ combinations of α and β, while the remaining methods only consider N moves.

In our first experiment, we initialized all variables to the first state and ran each method until the energy did not change between iterations. We used the truncation described in §4.2 for problems that did not satisfy (7) for all triplets of states. In Table 2, we show the energy of the local minima obtained divided by the energy of the local minimum with respect to α-expansion moves. In this table, a value of 1 indicates that the energy was identical to the energy obtained by α-expansion moves, and we use 1.0000 if the energy is close but not identical. In this experiment, the new moves with all β obtained the lowest energy on 6 of the 7 data sets, and strictly so in 4 of these cases (the exceptions were *Teddy* where using a random β lead to a lower score, and the montage data sets where other methods reached the same energy). Among the remaining methods, the more computationally efficient strategy of simply setting $\beta = \alpha + 1$ obtained the lowest energy on 5 of the 7 data sets.

In our second experiment we used the local minimum with respect to α-expansion moves as the initialization, and tested whether the methods that only consider N moves could escape from this local minimum. Table 3 shows the energies of the local optima obtained with this initialization. Here, we see that even with these simple choices of β that the new moves are able to escape the local minimum with respect to α-expansions for 5 out of the 7 data sets. In this experiment choosing $\beta = \alpha + 1$ gave the lowest energy on all data sets, and gave a strictly lower energy on 4 of the 7. Selecting $\beta = \alpha + 1$ is somewhat intuitive, the moves can

Table 2: Energy of local minima with respect to iterative descent methods beginning from all variables set to state 1, divided by the energy of the local minima with respect to α-expansion moves.

Name	$\alpha\beta$-Swap	α-Expansion	Random β	$\beta = \alpha - 1$	$\beta = \alpha + 1$	All β
Family	1.0203	1	0.9998	1	0.9998	0.9998
Pano	1.3182	1	1.0006	1	1	1
Tsukuba	1.0315	1	1.0012	1	1.0000	1.0000
Venus	1.8561	1	1.0015	0.9992	0.9979	0.9968
Teddy	1.0037	1	0.9998	1	1.0007	0.9999
Penguin	1.1283	1	1.0037	0.9936	0.9793	0.9758
House	0.7065	1	0.7841	0.9973	0.7038	0.7032

Table 3: Relative energy of local minima for different choices of β when initializing with a local minima with respect to α-expansions.

Name	Random β	$\beta = \alpha - 1$	$\beta = \alpha + 1$
Family	0.9998	1	0.9998
Pano	1	1	1
Tsukuba	1	1	1
Venus	1.0000	0.9992	0.9979
Teddy	1	1	0.9999
Penguin	0.9998	0.9902	0.9775
House	0.8050	0.9971	0.7038

be viewed as 'prematurely' expanding the next value of α into the region occupied by the current value of α. Although an improved configuration was found on at least one data set for each of the three tasks, the largest improvements were seen in the image restoration/inpainting tasks. This is likely due to the 256 possible labels, which is much larger than the other tasks. In Figure 5, we show the restoration of the *house* data where the largest improvement was observed. In this figure we see a noticeable difference between the local minima with respect to the different moves. This visual difference is also reflected quantitatively; the new moves lead to a reconstruction error (ℓ_1-norm distance to the original image) that is approximately two-thirds that of the local minimum with respect to α-expansions (reduced from 1303537 to 911207).

6   Discussion

There have been a variety of other generalizations of $\alpha\beta$-swaps and α-expansions proposed in the literature, and an exhaustive list of this literature would be outside our scope. However, among the generalizations we are aware of these other moves either (i) require additional assumptions, or (ii) can not be solved in polynomial-time. For example, range-swap and range-expansion moves generalize $\alpha\beta$-swaps and α-expansions (respectively), but these can not be

computed in polynomial-time without further assumptions (Veksler, 2007; Gould et al., 2009; Kumar et al., 2011). Kolmogorov and Rother (2007) consider implementing minimum-cut methods wihout restrictions on the energy functions, but in general their algorithm can not compute the optimal move. Lempitsky et al. (2010) consider a general class of moves called fusion moves. These moves consider two full configurations of the variables x and z, and the set of moves are of the form

$$y_i \leftarrow \left\{ x_i \text{ or } z_i. \right.$$

That is, we can replace an arbitrary number of elements of x by the corresponding elements from some alternative configuration z. The α-expansion β-shrink moves proposed here are a special case of this type of move, where z_i is set to β if $x_i = \alpha$ and z_i is set to α otherwise. However, while Proposition 4 shows that this special case is solvable in polynomial-time under condition (7), Lempitsky et al. (2010) do not give restrictions that would allow an optimal general fusion move to be solved in polynomial time. Indeed, because of the generality of fusion moves, any such restriction would need to be much more restrictive than (7).

Although in this paper we have focused on the case of minimizing unary and pairwise energies, it is possible to apply $\alpha\beta$-swaps and α-expansions to certain classes of higher-order energies (Kohli et al., 2007). We expect that the new moves discussed in this paper can also be extended to these scenarios, and indeed we believe that the new moves could be used in place of α-expansions in any of the many applications where these moves are currently used. Finally, we note that the implementation of the new moves will be made available online.
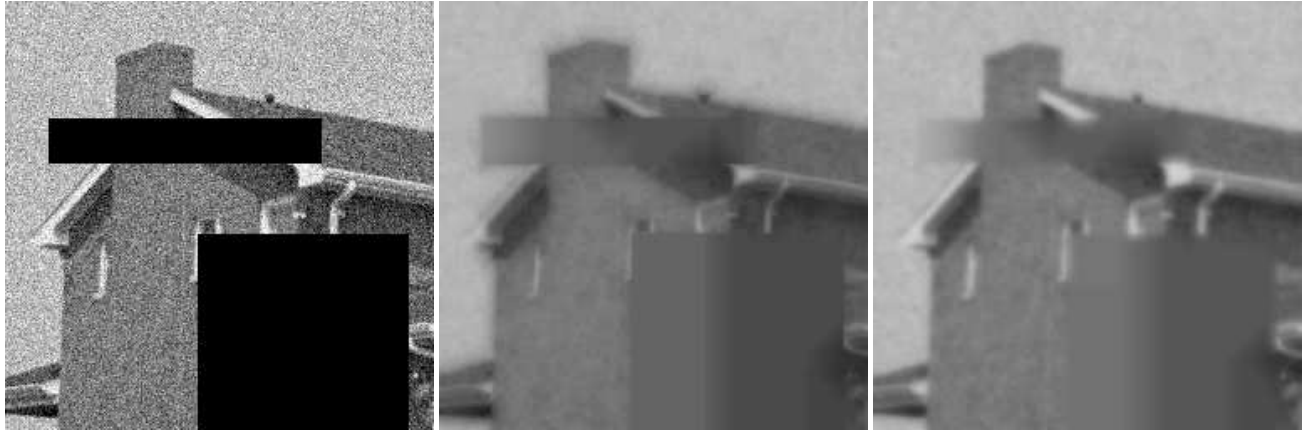
Acknowledgements

Figure 2: From left to right: Initial degraded image with missing areas, local minimum of energy function with respect to α-expansions, improved local minimum found by α-expansion β-shrink moves with $\beta = \alpha + 1$.

References

J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society: Series B*, 48(3):259–302, 1986.

Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. *IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

Y. Boykov, O. Veksler, and R. Zabih. Fast Approximate Energy Minimization via Graph Cuts. *IEEE International Conference on Computer Vision*, 1999.

A. Delong and Y. Boykov. A scalable graph-cut algorithm for nd grids. *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

S. Gould, F. Amat, and D. Koller. Alphabet SOUP: A framework for approximate energy minimization. *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

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

P. Hammer. Some network flow problems solved with pseudo-boolean programming. *Operations Research*, 13 (3):388–399, 1965.

H. Ishikawa. Exact optimization for Markov random fields with convex priors. *IEEE Transacations on Pattern Analysis and Machine Intelligence*, 25(10):1333–1336, 2003.

P. Kohli and P. Torr. Dynamic graph cuts for efficient inference in markov random fields. *IEEE Transacations on Pattern Analysis and Machine Intelligence*, 29(12): 2079–2088, 2007.

P. Kohli, M. Kumar, and P. Torr. P3 & beyond: Solving energies with higher order cliques. *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques.* The MIT Press, 2009.

V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts-a review. *IEEE Transacations on Pattern Analysis and Machine Intelligence*, 29 (7):1274–1279, 2007.

V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *European Conference on Computer Vision*, 2002.

M. Kumar, O. Veksler, and P. Torr. Improved moves for truncated convex models. *The Journal of Machine Learning Research*, 12:31–67, 2011.

V. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion Moves for Markov Random Field Optimization. *IEEE Transacations on Pattern Analysis and Machine Intelligence*, 32(8):1392–1405, 2010.

C. Rother, S. Kumar, V. Kolmogorov, and A. Blake. Digital tapestry. *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

N. Schraudolph and D. Kamenetsky. Efficient exact inference in planar Ising models. *Advances in Neural Information Processing Systems*, 2009.

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. *IEEE Transacations on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080, 2008.

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

O. Veksler. Graph cut based optimization for MRFs with truncated convex priors. *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.