The Twelfth IEEE International Conference on Computer Vision, Kyoto, Japan. September 27 - October 4, 2009.

Higher-Order Gradient Descent by Fusion-Move Graph Cut

Hiroshi Ishikawa

Nagoya City University
Department of Information and Biological Sciences
1 Yamanohata Mizuho, Nagoya 467-8501, Japan
hfs@ieee.org

Abstract

Markov Random Field is now ubiquitous in many formulations of various vision problems. Recently, optimization of higher-order potentials became practical using higher-order graph cuts: the combination of i) the fusion move algorithm, ii) the reduction of higher-order binary energy minimization to first-order, and iii) the QPBO algorithm. In the fusion move, it is crucial for the success and efficiency of the optimization to provide proposals that fits the energies being optimized. For higher-order energies, it is even more so because they have richer class of null potentials. In this paper, we focus on the efficiency of the higher-order graph cuts and present a simple technique for generating proposal labelings that makes the algorithm much more efficient, which we empirically show using examples in stereo and image denoising.

1. Introduction

Many problems in computer vision such as segmentation, stereo, and image restoration are often formulated as optimization problems involving inference of the maximum a posteriori solution of a probability distribution defined by Markov Random Fields (MRFs)[1]. Typically, the problem is defined as finding a labeling of pixels that minimizes a function on the space of labeling, called the energy. Such optimization schemes are now ubiquitous in vision, largely owing to the success of optimization techniques such as graph cuts[3, 9, 14], belief propagation[5, 21], and tree-reweighted message passing[13].

Recently, optimization of higher-order potentials became practical using what we might call the *higher-order graph cuts*: i.e., a combination of the following three techniques.

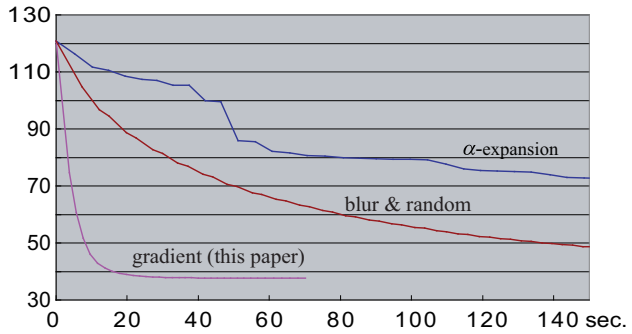

Figure 1. Minimization of a third-order energy for denoising of an image. The vertical axis is the energy and the horizontal axis is the time elapsed for optimization. The three plots are for the same energy discussed in 4.2. Alpha expansion failed to achieve the same level of minimum energy. Proposals by pixel-wise random label and blurring is used in [10]. The new technique by gradient descent proposal achieves the same level of minimum energy much faster.

i) The fusion move algorithm[18], which is a generalization of the α-expansion[3] algorithm that iteratively fuses the current labeling and a proposed labeling using binary optimization.

ii) The reduction of higher-order binary enery minimization to first-order. A method for reducing second-order energies[6, 14] has been known for some time, and its generalization[10] to general order was found recently.

iii) The QPBO algorithm[1, 2, 7], which allows the minimization of non-submodular binary problems.

The combination was first proposed by Woodford et al.[27] when they demonstrated a second-order prior for stereo, using the reduction[6, 14] then available. The more recent reduction[10] for energies of even higher order made the framework potentially more useful.

[1] The notion of conditional random field (CRF) is increasingly popular in the vision literature. A CRF contains both hidden and observed variables; when the values of the observed variables are fixed, the rest of the model on the hidden variables is an MRF. In the context of this paper, there is not much difference between MRF and CRF, since the observed data is assumed to be fixed before optimization.

In this paper, we focus on the *efficiency* of this algorithm. Specifically, we introduce a technique that makes it much faster (see Figure 1) by moving the current labeling by the gradient of the energy on the space of labelings to produce a proposal. We also show that using only a part of the energy for taking gradient can also be effective. We demonstrate it through experiments with energies of order two and three.

In the next section, we continue with an overview, explaining the relevant techniques. In section 3, we formulate the higher-order graph cuts in more detail and present the new technique for generating the proposal labelings. In section 4, we report the results of experiments showing the effectiveness of the technique using two examples, after which we conclude.

2. Preliminaries

In this section, we give an overview of the recent developments in higher-order energy minimization, which we then formally define. Then we explain the algorithms that the new method in this paper is directly related.

2.1. Higher-Order Energies

As the newer energy minimization techniques came to be widely used, the limitations of the energies readily optimizable by these algorithms came to be more salient. As pointed out by Meltzer et al.[21] after they introduced an algorithm that can find the global minimum of certain energy functions, even global minimum of an energy does not solve many problems if the energy itself does not reflect the given vision problem well enough. In particular, because of the lack of efficient general algorithms to optimize energies with higher-order interactions, in most applications energies are represented in terms of unary and pairwise clique potentials, with a few exceptions that consider triples[4, 14].

Higher-order energies can model more complex interactions and reflect the natural statistics better. This has been long realized[11, 22, 24], but with the success of energy optimization methods, there is a renewed emphasis on the necessity of an efficient way to optimize MRFs of higher-order. Since the algorithms mentioned above can only optimize energies with unary and pairwise clique potentials, there have been efforts to expand them to higher-order energies. For instance, belief propagation variants[17, 23] have been introduced to do inference based on higher-order clique potentials. Kohli et al.[12] extend the class of energy functions for which the optimal α-expansion moves can be computed in polynomial time. Komodakis and Paragios[16] employ a master-slave decomposition framework to solve a dual relaxation to the MRF problem. Rother et al.[26] use a soft-pattern-based representation of higher-order functions that may for some energies lead to very compact first-order functions with small number of non-submodular terms, as well as addressing the problem of transforming general multi-label functions into quadratic ones.

2.2. Minimization Problem

The energy minimization problem considered here is as follows. Let V be a set of pixels and L a set of labels, both finite sets. Also, let $\mathscr{C}$ be a set of subsets of V. We call an element of $\mathscr{C}$ a clique. The energy $E(X)$ is defined on the space L^V of labelings $X : V \to L$, i.e., an assignment of a label $X_v \in L$ to each pixel $v \in V$. The crucial assumption is that $E(X)$ is decomposable into a sum

$$E(X) = \sum_{C \in \mathscr{C}} f_C(X_C), \tag{1}$$

Here, for a clique $C \in \mathscr{C}$, $f_C(X_C)$ expresses a function f_C that depends only on labels assigned by X to the pixels v in C. We denote the set of labelings on the clique C by L^C; thus $X_C \in L^C$ and $f_C : L^C \to \mathbb{R}$. The cliques define a form of generalized neighborhood structure. In the case where all cliques consist of one or two pixels, it can be thought of as an undirected graph $G = (V, E)$, where the set of cliques are divided into the set of all singleton subsets of V and the set E of pairs of pixels, i.e., edges. In the general case, the pair $(V, \mathscr{C})$ can be thought of as a hypergraph. Then the Markov Random Field of first order is written as

$$E(X) = \sum_{v \in V} f_v(X_v) + \sum_{\{u,v\} \in E} f_{uv}(X_{uv}). \tag{2}$$

Although it is a little confusing, an n-th order MRF is the one that has cliques of size up to $n + 1$. Thus, a second-order MRF can have cliques containing three pixels, and an energy of the form

$$\sum_{\{v\} \in C} f_{\{v\}}(X_{\{v\}}) + \sum_{\{u,v\} \in C} f_{\{uv\}}(X_{\{uv\}}) + \sum_{\{u,v,w\} \in C} f_{\{uvw\}}(X_{\{uvw\}}). \tag{3}$$

As we have already done above, we sometimes abuse the notation slightly and write X_{uv} instead of $X_{\{u,v\}}$, and so on.

2.3. Graph Cuts and Move-Making Algorithms

Currently, one of the most popular optimization techniques in vision is α-expansion[3]. It starts from an initial labeling and iteratively makes a series of moves by solving a binary optimization problem with an s-t mincut algorithm, which can globally optimize a class of binary energy potentials called submodular functions[14]. Here, we explain it and its extension, the fusion move algorithm.

The α-expansion Algorithm The α-expansion algorithm starts with an arbitrary labeling X and iteratively makes a move, i.e., a change of labeling. In one iteration, the algorithm changes the labeling so that the energy becomes

smaller or at least stays the same. The move is decided by either keeping the original label or replacing it with a globally fixed label α at each pixel. Thus, the "area" that has the label α can only expand; hence the name α-expansion. The choice of either leaving the label the same or changing it to α at each pixel defines a binary labeling problem, using as its energy the original energy after the move. By minimizing the binary energy, the move that reduces the energy most is chosen. When the binary problem is submodular, it can be solved globally by the minimum-cut algorithm. By visiting all labels α in some order, and repeating it, $E(X)$ is approximately minimized. The algorithm also has a guarantee on how close it can approach the global minima.

The Fusion Move The fusion move[18, 19] is a simple generalization of α-expansion: in each iteration, define the binary problem as the choice at each pixel between two arbitrary labelings, instead of between the current label and a fixed label α. It seems so simple and elegant that one may wonder why the move-making algorithm was not formulated this way from the beginning. The answer is simple: it is because fusion moves are non-submodular in general, whereas in the case of α-expansion, the submodularity can be guaranteed by some simple criterion. For instance, if the pairwise energy is a metric, each α-expansion is guaranteed to be submodular. It is only because of the emergence of the QPBO/roof-duality optimization below that we can now consider the general fusion move. Another special case that has a guarantee of submodularity is α-β swap, which allows those moves that, at each pixel, leave the current label unchanged or swap the two fixed labels α and β.

QPBO Separately, there is a recent (at least to the vision community) innovation that is important to the move-making graph-cut algorithms: the optimization of binary energies saw an advance that allows the optimization of non-submodular functions. This method by Boros, Hammer and their co-workers [1, 7, 2] is variously called QPBO[15] or roof-duality[25] in the vision literature. If the function is submodular, QPBO is guaranteed to find the global minimum. Even if it is not submodular, QPBO returns a partial solution assigning either 0 or 1 to some of the pixels, leaving the rest unlabeled. The algorithm guarantees that the partial labeling is a part of a global minimum labeling. This has a crucial impact on the move-making algorithms since the choice of the move in each iteration depends on binary-label optimization. In particular, QPBO has an "autarky" property[15]: if we take any labeling and "overwrite" it with a partial labeling obtained by QPBO, the energy for the resulting labeling is not higher than that for the original labeling. This lets us ensure that energy does not increase in move-making: we just leave the label unchanged at those pixels that are not labeled by QPBO. In the context of optimizing higher-order potentials, it means that some limitations that prevented the use of these algorithms for higher-order functions can possibly be overcome.

2.4. Higher-Order Graph Cut

Now, the development of the two algorithms above—fusion move and QPBO—has an important implication in optimization of higher-order energies. For second order potentials (i.e., those with cliques having up to three pixels), a reduction technique that can reduce them into pairwise potentials has been introduced by Kolmogorov and Zabih[14] and later reformulated by Freedman and Drineas[6]. However, for the result of the reduction to be minimized with the popular techniques such as graph cuts, it must be submodular. This requirement made its actual use quite rare, if not nonexistent.

Thanks to the QPBO technique, now we can think of reducing higher-order potentials into pairwise ones with a hope that at least part of the solution can be found. Although it is not a solution to every problem, the QPBO technique often allows solving non-submodular problem approximately by giving a large enough part of the globally optimal solution to be used in a move-making algorithm to iteratively improve the solution. Moreover, we recently introduced[10] a technique that extends the reduction in [6] and [14] to general higher-order energies. It is a simple technique that can convert the minimization problem of any higher-order binary energy to that of a first order energy.

In the same paper, we also obtained a result suggesting that the use of fusion move as opposed to α-expansion is more important in higher-order energies. This may be because higher-order energies have richer class of null (or almost null) potentials. The proposals in α-expansion are constant labelings; and α-expansion works well exactly with those energies, such as the Potts potential, that heavily favors piecewise-constant solutions. For higher-order energies, it is crucial for the success and efficiency of the optimization to provide proposals that fits the energies being optimized.

The combination, which we might call the higher-order graph cuts, was first introduced by Woodford et al.[27], where they show that second-order smoothness priors can be used for stereo reconstruction by introducing an interesting framework. It integrates existing stereo algorithms, which may sometimes be rather *ad hoc*, and combines their results in a principled way. In the framework, an energy is defined to represent the global tradeoff of various factors. The energy is minimized using the fusion move algorithm, in which the various existing algorithms are utilized to generate the proposals in each iteration. In particular, this allows the powerful segmentation-based techniques to be combined with the global optimization methods. In this way, the framework allows the integration of energy formu-

lation and any other stereo method that produces disparity map.

3. Gradient-Descent Fusion Move

In this section, we first formulate the higher-order graph cuts in more detail and then present the new technique for generating the proposal labelings.

3.1. Higher-Order Graph Cut Algorithm

The algorithm solves the minimization problem explained in 2.2 by minimizing the energy (1).

It maintains the current labeling X. In each iteration, the algorithm fuses X and a proposed labeling $P \in L^V$ by minimizing a binary energy. For instance, in the α-expansion algorithm, the proposal is a constant labeling with label α everywhere. Here, it can be any labeling and how it is prepared is problem-specific, as is how X is initialized at the beginning.

Let us denote the set $\{0, 1\}$ of binary labels by $\mathbb{B}$. The following binary energy is minimized every iteration. We consider a binary labeling variable $Y \in \mathbb{B}^V$. It consists of a binary variable $Y_v \in \mathbb{B}$ for each pixel $v \in V$ that indicates the choice of the value that X_v will have at the end of the iteration. That is, $Y_v = 0$ if X_v is to remain the same and $Y_v = 1$ if X_v is to change to the proposed label P_v.

Let us denote by $F_C^{X,P}(\beta) \in L^C$ the labeling on clique C that X will have if the value Y_C of Y on C is $\beta \in \mathbb{B}^C$:

$$\left(F_C^{X,P}(\beta)\right)_v = \begin{cases} X_v & \text{if } \beta_v = 0 \\ P_v. & \text{if } \beta_v = 1 \end{cases} \qquad (v \in C) \tag{4}$$

With this notation, we define a binary energy

$$\mathscr{E}(Y) = \sum_{C \in \mathscr{C}} \sum_{\beta \in \mathbb{B}^C} f_C\left(F_C^{X,P}(\beta)\right) \theta_C^\beta(Y_C), \tag{5}$$

where $\theta_C^\beta(Y_C)$ is a polynomial of degree $|C|$ defined by

$$\theta_C^\beta(Y_C) = \prod_{v \in C} \{\beta_v Y_v + (1 - \beta_v)(1 - Y_v)\}, \tag{6}$$

which is 1 if $Y_C = \beta$ and 0 otherwise.

The polynomial $\mathscr{E}(Y)$ is then reduced into a quadratic one, i.e., a first-order MRF, using the technique described in [10] or its predecessor [6, 14]. We then minimize the energy using the QPBO algorithm, obtaining a partial labeling. For each pixel v that is labeled 1, we update X_v to P_v, leaving it unchanged for other pixels. We iterate the process until some convergence criterion is met.

3.2. Proposal by Gradient Descent

The successful use and efficiency of the fusion move algorithm crucially depends on the choice of the proposal labeling P. We can consider the α-expansion as a special case where the proposal labeling is the constant labeling that assigns the same label to every pixel.

The contribution of this paper is the simple but useful observation that often the energy $E(X)$ or its part can be differentiated, and then we can move the current labeling by the gradient of the energy in order to generate the proposal labeling.

$$P = X - \eta \operatorname{grad} E(X) \tag{7}$$

$$P_v = X_v - \eta \frac{\partial E}{\partial X_v}. \tag{8}$$

That is, we can generate the proposal labeling in each iteration by using a simple gradient descent. We found empirically that, unlike in the case of ordinary gradient descent, we can omit some of the energy and still get an efficient result. For instance, we can use only the prior, higher-order part of the energy to get the gradient and move the labeling in that direction, with very fast decrease in energy at the beginning of the minimization process. If we do that in ordinary gradient descent, there is usually no guarantee that energy even decreases. However, the fusion move is always guaranteed to decrease, or at least not increase, the energy; so we can do rough things such as this. Another benefit of the protection by the graph cuts is that we can take a very large step compared to ordinary gradient descent.

In the next section, we give concrete examples and also demonstrate the efficiency of the method.

4. Experiments

We show the effectiveness of the gradient descent proposal generation in higher-order graph cuts with two examples. One is the second-order stereo[27] that uses the curvature term. The other is a third-order denoising example.

4.1. Second-order Stereo Prior

Woodford et al.[27] showed that second-order smoothness priors can be beneficial to stereo reconstruction. Their algorithm is the higher-order graph-cut algorithm explained in 3.1 with the following special case of (1) as the energy:

$$E(X) = E_{\mathrm{D}}(X) + E_{\mathrm{P}}(X) \tag{9}$$

$$= \sum_{v \in V} f_v(X_v) + \sum_{C \in \mathscr{C}_{\mathrm{P}}} f_C(X_C). \tag{10}$$

Here, the set $\mathscr{C}$ of cliques is divided into the set of singleton cliques $\mathscr{C}_{\mathrm{D}} = \{\{v\} \mid v \in V\}$ and the set $\mathscr{C}_{\mathrm{P}}$ of cliques with three pixels.

The prior approximates the second derivative; thus each clique C in $\mathscr{C}_{\mathrm{P}}$ consists of three consecutive pixels $C = (u, v, w)$ as shown in Figure 2, and the energy term is a truncated absolute second difference of the disparity:

$$f_C(X_C) = W_C \min\left(\sigma_s, |X_u - 2X_v + X_w|\right), \tag{11}$$

where $\sigma_s > 0$ is a constant cut-off value for robustness and W_C is a weight for each clique. The gradient of the prior $E_P(X)$ has, as the component for each pixel v, the partial derivative by X_v:

$$(\mathrm{grad}E_P(X))_v = \sum_{C\in\mathscr{C}} \frac{\partial f_C(X_C)}{\partial X_v} = \sum_{C,v\in C} \frac{\partial f_C(X_C)}{\partial X_v}. \quad (12)$$

The RHS is the sum of the terms for (in general) six cliques C that contain v (Figure 3). Each term depends on the sign and the absolute values of the second difference, e.g.,:

$$\frac{\partial f_C(X_C)}{\partial X_v} = \begin{cases} W_C & \text{if } C=(v,u,w), 0 < X_v - 2X_u + X_w < \sigma_s \\ 2W_C & \text{if } C=(u,v,w), -\sigma_s < X_u - 2X_v + X_w < 0 \\ 0 & \text{if } C=(u,v,w), \sigma_s < X_u - 2X_v + X_w \\ etc. & \end{cases} \quad (13)$$

We omit the rest since it is clear that they can be computed easily.

In the first phase of their fusion-move optimization, Woodford et al.[27] use the result of the segmentation-based stereo with different degree of coarseness as proposals. They first use a local window matching process to generate an approximate disparity map. Then they use 14 sets of parameters in total and two segmentation algorithms to produce segmentations of different degree of coarseness, ranging from highly undersegmented to highly oversegmented. For each segment in each segmentation, LO-RANSAC is used to find the plane that produces the greatest number of inlying correspondences by the approximate disparity and set the proposal so that all the pixels in the segment lie on that plane. In the second phase, α-expansion-like constant proposals are used with uniformly random disparity.

Experiments Then, in the third phase, they use the result of the first and second phases as two of the six rotating proposals. For the other four of the six proposals, they use the proposal generated by a smoothing operation on the current disparity map. Curiously, they say the smoothing proposal "can be viewed as a proxy for local methods such as gradient descent," but do not seem to have actually tested the simple gradient descent for generating proposals.

We use the gradient descent proposal

$$P_v = X_v - \eta\,(\mathrm{grad}E_P(X))_v \quad (14)$$

instead of the smoothing proposals and compare the result of the third phase with the original one. Table 1 and 2 show that the technique yields a result with the comparable quality and energy almost twice as fast.

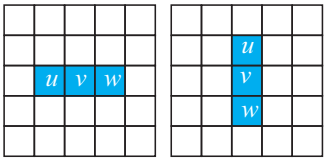

Figure 2. The second-order cliques used in Woodford et al.[27].

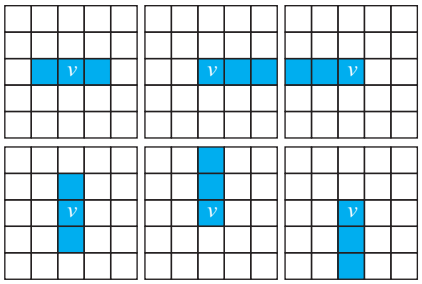

Figure 3. Six cliques that contains a particular pixel in [27].

Table 1. Second-order stereo results for the "cones" stereo pair.

	Bad pixels	Final Energy	Time
Original[27]	5.82%	2.84×10^{10}	2153 sec.
$\eta = 0.003$	5.78%	2.84×10^{10}	1215 sec.
$\eta = 0.01$	7.06%	2.86×10^{10}	740 sec.

Table 2. Second-order stereo results for the "teddy" stereo pair.

	Bad pixels	Final Energy	Time
Original[27]	6.59%	2.60×10^{10}	1630 sec.
$\eta = 0.001$	6.55%	2.61×10^{10}	1188 sec.
$\eta = 0.003$	6.67%	2.62×10^{10}	524 sec.

4.2. Denoising by Third-order Field of Experts

In [10], we use an image denoising problem to test the higher-order graph cuts with a third-order energy, which is made possible by the new reduction introduced in the paper. The image restoration scheme uses the recent image statistical model called the Fields of Experts (FoE) by Roth and Black[24], which captures complex natural image statistics beyond pairwise interactions by providing a way to learn an image model from natural scenes. FoE has been shown to be highly effective, performing well at image denoising and image inpainting using a gradient descent algorithm. Similar to its predecessor, the Product of Experts model[8], the FoE model represents the prior probability of an image as the product of several student-T distributions:

$$p(X) \propto \prod_C \prod_{i=1}^{K} \left(1 + \frac{1}{2}(J_i \cdot X_C)^2\right)^{-\alpha_i}. \quad (15)$$

where C runs over the set of all $n\times n$ patches in the image, and J_i is an $n\times n$ filter. The parameters J_i and α_i are learned

from a database of natural images.

In the simple image restoration problem, we are given a noisy image N made by adding a Gaussian i.i.d. noise with known standard deviation σ and find the maximum a posteriori estimation according to the prior model (15). The prior gives rise to a third-order MRF, with clique potentials that depend on up to four pixels. The likelihood of noisy image N given the true image X is assumed to be

$$p(N|X) \propto \prod_{v\in V} \exp\left(-\frac{(N_v - X_v)^2}{2\sigma^2}\right). \tag{16}$$

The energy in [10] is of the form (1). Specifically, it is:

$$E(X) = E_{\mathrm{D}}(X) + E_{\mathrm{P}}(X) \tag{17}$$

$$E_{\mathrm{D}}(X) = \frac{1}{2\sigma^2}\sum_{v\in V}(N_v - X_v)^2, \tag{18}$$

$$E_{\mathrm{P}}(X) = \sum_{C\in\mathscr{C}_{\mathrm{P}}}\sum_{i=1}^{3}\alpha_i \log\left(1 + \frac{1}{2}(J_i \cdot X_C)^2\right), \tag{19}$$

where the set $\mathscr{C}_{\mathrm{P}}$ of cliques consists of all 2×2 patches in the image.

Experiments In [10], similarly to the stereo example[27], the proposed image is one of the following two, alternating each iteration:

i) a uniform random image created every time it is used,

ii) a blurred image, which is made every 30 iterations by blurring the current image with a Gaussian kernel ($\sigma = 0.5625$).

We compared the results by this original algorithm with the gradient descent fusion proposal

$$P_v = X_v - \eta\,(\mathrm{grad}E(X))_v\,. \tag{20}$$

From the energy (18) and (19), the partial derivative of the energy is

$$\frac{\partial E(X)}{\partial X_v} = \frac{1}{\sigma^2}\sum_{v\in V}(X_v - N_v) + \sum_{C\in\mathscr{C}_{\mathrm{P}}}\sum_{i=1}^{3}\alpha_i \frac{J_i^{(v)}}{\left(1 + \frac{1}{2}(J_i \cdot X_C)^2\right)},$$

where $J_i^{(v)}$ denotes the component of the filter J_i that corresponds to the pixel v.

Table 3 shows the energy and the time for four example noisy images with $\sigma = 20$ made from four images in the Berkeley segmentation database [20], grayscaled and reduced in size. Figure 4 shows the result for one of the example images (test003). The quality of the two denoising results are comparable, but the new result was produced almost 20 times faster. Figure 1 on the first page shows the plot of the energy for the original and our proposal method, as well as the α-expansion, for the example image test001. All experiments used a 2.33GHz Xeon E5345 processor.

Table 3. Third-order image restoration results.

	test001	test002	test003	test004
Original[10] E	37769	25030	29805	27356
Original time	1326 sec.	1330 sec.	1305 sec.	1290 sec.
This paper E	38132	24831	29683	27354
This paper time	71 sec.	81 sec.	67 sec.	79 sec.

5. Conclusion

We suggested in [10] that the use of fusion moves as opposed to α-expansion is more important in higher-order energies, because they have richer class of null potentials. The proposals in α-expansion are constant labelings; and α-expansion works well exactly with those energies, such as the Potts potential, that heavily favors piecewise-constant solutions. The message this paper brings is that, for fusion moves with higher-order energies, it is crucial for the success and efficiency of the optimization to provide proposals that fit the energies being optimized.

For each application, a mix of various strategies will be probably necessary. In this paper, we point out a simple technique: we generate proposal labelings in fusion moves by moving the current labeling by the gradient of the energy. It is not really gradient descent as i) the gradient of even a part of the energy (e.g. the prior in the stereo experiment) can be used to speed up the fusion move, and ii) unlike ordinary gradient descent, the descent step can safely be made quite large, which speeds up the optimization especially in the early stage. These can safely be done only because the actual move is "guarded" against increasing the energy by graph cuts. Although it is simple and seems obvious, the technique makes the fusion move algorithm much more efficient in the case of higher-order potential.

Acknowledgments

This work was partially supported by the Kayamori Foundation and the Grant-in-Aid for Scientific Research 19650065 from the Japan Society for the Promotion of Science.

References

[1] E. Boros, P. L. Hammer, and X. Sun. "Network Flows and Minimization of Quadratic Pseudo-Boolean Functions." RUTCOR Research Report, RRR 17-1991, May 1991. 1, 3

[2] E. Boros, P. L. Hammer, and G. Tavares. "Preprocessing of Unconstrained Quadratic Binary Optimization." RUTCOR Research Report, RRR 10-2006, April 2006. 1, 3

[3] Y. Boykov, O. Veksler, and R. Zabih. "Fast Approximate Energy Minimization via Graph Cuts." *IEEE TPAMI* **23**:1222–1239, 2001. 1, 2

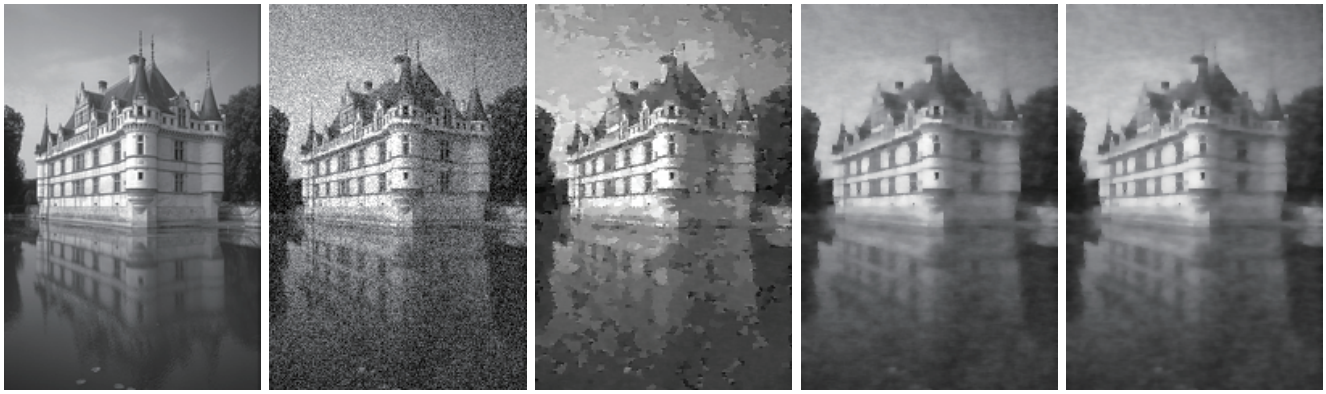

Figure 4. From left to right: original image (test003), noise-added image (Gaussian, $\sigma = 20$), denoised using the α expansion proposal, the proposals in [10], and using the new technique. The last two denoising results are comparable, but the new result was produced almost 20 times faster.

[4] D. Cremers and L. Grady. "Statistical Priors for Efficient Combinatorial Optimization Via Graph Cuts." In *ECCV2006*, III:263–274. 2

[5] P. Felzenszwalb and D. Huttenlocher. "Efficient Belief Propagation for Early Vision." *Int. J. Comp. Vis.* **70**:41–54, 2006. 1

[6] D. Freedman and P. Drineas. "Energy minimization via Graph Cuts: Settling What is Possible." In *CVPR2005* II:939–946. 1, 3, 4

[7] P. L. Hammer, P. Hansen, and B. Simeone. "Roof Duality, Complementation and Persistency in Quadratic 0-1 Optimization." *Mathematical Programming* **28**:121–155, 1984. 1, 3

[8] G. Hinton. "Training products of experts by minimizing contrastive divergence." *Neural Comp.*, **14**(7):1771–1800, 2002. 5

[9] H. Ishikawa. "Exact Optimization for Markov Random Fields with Convex Priors." *IEEE TPAMI* **25**(10):1333–1336, 2003. 1

[10] H. Ishikawa. "Higher-Order Clique Reduction in Binary Graph Cut." In CVPR2009. 1, 3, 4, 5, 6, 7

[11] H. Ishikawa and D. Geiger. "Rethinking the Prior Model for Stereo." In *ECCV2006*, III:526–537. 2

[12] P. Kohli, M. P. Kumar, and P. H. S. Torr. "$\mathcal{P}^3$ & Beyond: Move Making Algorithms for Solving Higher Order Functions." *IEEE TPAMI*, **31**(9):1645–1656, 2009. 2

[13] V. Kolmogorov. "Convergent Tree-Reweighted Message Passing for Energy Minimization." *IEEE TPAMI* **28** (10): 1568–1583, 2006. 1

[14] V. Kolmogorov and R. Zabih. "What Energy Functions Can Be Minimized via Graph Cuts?" *IEEE TPAMI* **26**(2):147–159, 2004. 1, 2, 3, 4

[15] V. Kolmogorov and C. Rother. "Minimizing Non-submodular Functions with Graph Cuts—A Review." *IEEE TPAMI* **29**(7):1274–1279, 2007. 3

[16] N. Komodakis and N. Paragios "Beyond Pairwise Energies: Efficient Optimization for Higher-order MRFs." In *CVPR2009*. 2

[17] X. Lan, S. Roth, D. P. Huttenlocher, and M. J. Black. "Efficient Belief Propagation with Learned Higher-Order Markov Random Fields." In *ECCV2006*, II:269–282. 2

[18] V. Lempitsky, C. Rother, and A. Blake. "LogCut - Efficient Graph Cut Optimization for Markov Random Fields." In *ICCV2007*. 1, 3

[19] V. Lempitsky, S. Roth, and C. Rother. "FusionFlow: Discrete-Continuous Optimization for Optical Flow Estimation." in *CVPR2008*. 3

[20] D. Martin, C. Fowlkes, D. Tal, and J. Malik. "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics." In *ICCV2001*, pp. 416–423. 6

[21] T. Meltzer, C. Yanover, and Y. Weiss. "Globally Optimal Solutions for Energy Minimization in Stereo Vision Using Reweighted Belief Propagation." In *ICCV2005*, pp. 428–435. 1, 2

[22] R. Paget and I. D. Longstaff. "Texture Synthesis via a Noncausal Nonparametric Multiscale Markov Random Field." *IEEE Trans. on Image Processing*, **7**(6):925–931, 1998. 2

[23] B. Potetz. "Efficient Belief Propagation for Vision Using Linear Constraint Nodes." In *CVPR2007*. 2

[24] S. Roth and M. J. Black. "Fields of Experts: A Framework for Learning Image Priors." In *CVPR2005*, II:860–867. 2, 5

[25] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. "Optimizing Binary MRFs via Extended Roof Duality." In *CVPR2007*. 3

[26] C. Rother, P. Kohli, W. Feng, and J. Jia. "Minimizing Sparse Higher Order Energy Functions of Discrete Variables." In *CVPR2009*. 2

[27] O. J. Woodford, P. H. S. Torr, I. D. Reid, and A. W. Fitzgibbon. "Global Stereo Reconstruction under Second Order Smoothness Priors." In *CVPR2008*. 1, 3, 4, 5, 6