GRSA: Generalized Range Swap Algorithm for the Efficient Optimization of MRFs

Kangwei Liu, Junge Zhang, Peipei Yang and Kaiqi Huang
Center for Research on Intelligent Perception and Computing (CRIPAC), National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA)

Markov Random Field (MRF) is an important tool and has been widely used in many vision tasks. Thus, the optimization of MRFs is a problem of fundamental importance. Recently, Veskler [6] and Kumar *et.al* [3] propose the range move algorithms, which are one of the most successful solvers to this problem. However, two problems have limited the applicability of previous range move algorithms: 1) They are limited in the types of energies they can handle (*i.e.* only truncated convex functions); 2) These algorithms tend to be very slow compared to other graph-cut based algorithms (e.g. α-expansion and $\alpha\beta$-swap [1]). To solve the problems, we propose a generalized range swap algorithm (GRSA) for efficient optimization of MRFs in this paper.

The GRSA optimizes the MRFs by a series of iterative moves, and it converges when there is no move can be found to decrease the MRF energy. We define $\mathcal{L}=\{0,\cdots,n\}$ to be the label set, and $\mathcal{L}_s=\{l_1,\cdots,l_m\}$ ($l_i<l_{i+1}$) to be a subset chosen from $\mathcal{L}$. Let $\mathcal{P}_l=\{p\in\mathcal{P}|f_p=l\}$ denote the set of vertices assigned label l, and $\mathcal{P}_\mathcal{S}=\{p\in\mathcal{P}|f_p\in\mathcal{L}_s\}$ denote the set of vertices whose labels belong to $\mathcal{L}_s$. Then, a move from f to f' is called a *range swap move* (RSM) on $\mathcal{L}_s$, if $\mathcal{P}'_\mathcal{S}=\mathcal{P}_\mathcal{S}$, and $\mathcal{P}'_l=\mathcal{P}_l$ for any label $l\notin\mathcal{L}_s$. To address the first problem, we extend the GRSA to arbitrary semimetric energies by restricting the chosen labels in each move so that the energy satisfies the following submodular condition [4] on the chosen subset.

Definition 1 *Given a pairwise potential $\theta(\alpha,\beta)$, we call $\mathcal{L}_s$ a **submodular set**, if it satisfies*

$$\theta(l_{i+1},l_j)-\theta(l_{i+1},l_{j+1})-\theta(l_i,l_j)+\theta(l_i,l_{j+1})\geq 0 \qquad (1)$$

for any pair of labels l_i, $l_j\in\mathcal{L}_s(1\leq i,j<m)$.

Furthermore, to feasibly choose the labels which satisfy the submodular condition, we provide a sufficient condition of the submodularity.

Theorem 1 *Given a pairwise function $\theta(\alpha,\beta)=g(x)$ ($x=|\alpha-\beta|$) on domain $X=[0,c]$, assume there is an interval $X_s=[a,b]$ ($0\leq a<b\leq c$) satisfying: (i) $g(x)$ is locally convex on $[a,b]$, and (ii) $a*(g(a+1)-g(a))\geq g(a)-g(0)$. Then $\mathcal{L}_s=\{l_1,\cdots,l_m\}$ is a submodular subset, if $|l_i-l_j|\in[a,b]$ for any pair of labels $l_i,l_j\in\mathcal{L}_s$.*

With Theorem 1, we can obtain a series of candidate submodular sets while given an arbitrary semimetric functions. The range swap move executed on any of these submodular sets can be exactly solved by computing the *st*-mincut problem.

For the second problem, previous range swap algorithms execute the set of all possible range moves $\mathcal{L}_{\alpha\beta}=\{\alpha,\alpha+1,\cdots,\beta\}$, where $|\alpha-\beta|=T$, and T is the truncated factor in a truncate convex function (*e.g.* $\theta=\min\{|f_p-f_q|,T\}$). However, there are many repeated labels in these moves. In practice, we find the requirement is sufficient to ensure the quality of solutions that *any pair of labels should be simultaneously considered once in one cycle of iterative moves*, *i.e.*, every vertex should have chance to swap its current label with any other labels. We dynamically obtain the iterative moves by solving a *set cover problem* (SCP) [2], which greatly reduces the number of moves during the optimization.

In a SCP, we are usually given an *universe* U of m elements, a collection of set $\mathcal{S}=\{S_1,...,S_k\}$ where $S_i\subseteq U$, and a cost function $c:\mathcal{S}\rightarrow\mathbf{R}$. The objective of the set cover problem is to find a cover $\mathcal{S}'\subseteq\mathcal{S}$ that covers all the element in U and minimizes the costs.

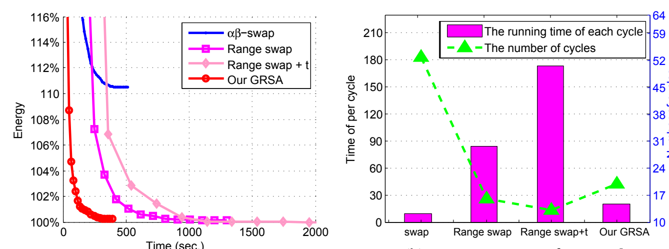
In the GRSA, let $\mathcal{L}_1,\mathcal{L}_2,\cdots,\mathcal{L}_k$ be the series of submodular sets and let $\mathcal{C}(\mathcal{L})=\{(0,1),(0,2),\cdots,(n-1,n)\}$ to be the set containing all the pairs of labels in $\mathcal{L}$. We define the universe $U=\mathcal{C}(\mathcal{L})$, and the collection of set $S_i=\mathcal{C}(\mathcal{L}_i)$. Therefore, the moves can be obtained by solving the following set cover problem:

$$\min\sum_{S_i\in\mathcal{S}'}c(S_i) \qquad \text{s.t.}\bigcup_{S_i\in\mathcal{S}'}S_i=U\,. \qquad (2)$$

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.

Algorithm 1 The Generalized Range Swap Algorithm

Input:
The label set $\mathcal{L}=\{0,\cdots,n\}$, and the pairwise function $\theta(\alpha,\beta)=g(x)$ ($x=|\alpha-\beta|$).
Initialization:
Find the series of submodular sets $\mathcal{L}_i$ with the total enumeration according to Theorem 1.
Get the collection of sets $\mathcal{S}=\{S_1,...,S_k\}$ where $S_i=\mathcal{C}(\mathcal{L}_i)$, and initialize $U=\mathcal{C}(\mathcal{L})$, $S_c:=\varnothing$.
Initialize the labeling f.
Iteration:
repeat
  while $S_c\neq U$ **do**
    Choose $S_i\in\mathcal{S}$, which minimizes the cost per element $\frac{c(S_i)}{|S_c\cup S_i|-|S_c|}$.
    Set $S_c:=S_i\cup S_c$ and $\mathcal{L}_s:=\mathcal{L}_i$ where $S_i=\mathcal{C}(\mathcal{L}_i)$.
    Get the new labeling $f'=\arg\min E(f)$ within the range swap move on $\mathcal{L}_s$.
    If $E(f')<E(f)$, set $f:=f'$.
  end while
until No moves can be found to decrease $E(f)$.
Output:
Return the labeling f.



(a) Penguin: Energy vs. Time

(b) Penguin: Time of per cycle vs. number of cycles

Figure 1: The results obtained on image restoration. (a) shows the energy obtained by different algorithms with running time on *penguin*. (b) shows the running time taken by every cycle of iterations in different algorithms, and the number of cycles that each algorithm takes to converge.

Although the SCP is an NP hard problem, fortunately, the greedy algorithm [5] can successfully achieve an approximate solution in polynomial time. Algorithm 1 describes the iterative process of the GRSA, where the moves are chosen by dynamically solving the SCP with the greedy algorithm.

We evaluate the GRSA on both synthetic data and real vision applications of image restoration and stereo matching. As illustrated in Figure 1, experiments show that the GRSA offers a great speedup over previous range swap algorithms (e.g. it can be at least 3-6 times faster than previous range swap methods, and sometimes even faster than $\alpha\beta$-swap), while it obtains competitive solutions.

[1] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *TPAMI*, 23(11), 2001.

[2] Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45:634–652, 1998.

[3] M Pawan Kumar and Philip HS Torr. Improved moves for truncated convex models. In *NIPS*, 2008.

[4] Dmitrij Schlesinger and Boris Flach. *Transforming an arbitrary minsum problem into a binary one*. TU, Fak. Informatik, 2006.

[5] Peter Slavĺk. A tight analysis of the greedy algorithm for set cover. *Journal of Algorithms*, 25, 1997.

[6] Olga Veksler. Graph cut based optimization for mrfs with truncated convex priors. In *CVPR*, 2007.