Generative Image Segmentation Using Random Walks with Restart

Tae Hoon Kim, Kyoung Mu Lee, and Sang Uk Lee

Dept. of EECS, ASRI, Seoul National University, 151-742, Seoul, Korea
thkim@diehard.snu.ac.kr, kyoungmu@snu.ac.kr, sanguk@ipl.snu.ac.kr

Abstract. We consider the problem of multi-label, supervised image segmentation when an initial labeling of some pixels is given. In this paper, we propose a new generative image segmentation algorithm for reliable multi-label segmentations in natural images. In contrast to most existing algorithms which focus on the inter-label discrimination, we address the problem of finding the generative model for each label. The primary advantage of our algorithm is that it produces very good segmentation results under two difficult problems: the *weak boundary problem* and the *texture problem*. Moreover, single-label image segmentation is possible. These are achieved by designing the generative model with the Random Walks with Restart (RWR). Experimental results with synthetic and natural images demonstrate the relevance and accuracy of our algorithm.

1 Introduction

Image segmentation is an important issue in computer vision. Specially, the segmentation of natural images is one of the most challenging issues. Two important difficulties of the segmentation in natural images are the *weak boundary problem* and the *texture problem*. The first problem is to find weak boundaries when they are parts of a consistent boundary. The second problem is to separate the texture in the highly cluttered image. In fact, such situations often arise in natural images. In these cases, the segmentations become ambiguous without user-provided inputs, and thus the supervised image segmentation approaches are often preferred. In this paper, we address the supervised image segmentation problem.

Recently, several supervised segmentation approaches have been proposed. There are three types of supervised segmentation algorithms according to the user inputs. The first type is that the segmentation is obtained based on pieces of the desired boundary, such as the intelligent scissors [1]. The second type is that an initial boundary that is closed to the desired boundary is given, such as Active Contour [2] and Level Set [3]. Finally, the third type is that the user provides an initial labeling of some pixels. We focus on the supervised image segmentation of the third type. One of the popular approaches is the Graph Cuts method (GC) [4] based on energy functionals which are minimized via discrete optimization techniques. The set of edges with minimum total weights is obtained via max-flow/min-cut energy minimization. Since GC treats this minimum cut criterion,

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part III, LNCS 5304, pp. 264–275, 2008.

it often causes *small cut* problem when the contrast is low or the number of seed pixels is small.

In [5], the geodesic distance from the seeds was used for image segmentation. By assigning each pixel the label with minimum distance, the segmentation is obtained. The geodesic distance between two pixels is simply defined as the smallest integral of a weight function over all paths. However, since it does not consider the global relationship between two pixels, it is not reliable to use the simple geodesic distance as the relevance measure between two pixels.

Another approach is the Random Walker image segmentation algorithm (RW) proposed by Grady [6]. After the *first arrival* probability that a random walker starting at a pixel first reaches one of the seeds with each label is computed, that pixel takes one label with maximum probability. It was shown in [6] that RW has better performance under difficult conditions than GC. However, the *first arrival* probability defined in [6] has some limitations. Since a random walker starting at a pixel must first arrive at the border of pre-labeled region, it only considers the local relationship between the pixel and that border. Therefore, the information of seeds inside the pre-labeled region is ignored in absence of higher-order interactions. Also, this probability depends on the number of seeds. If the seeds with only one label numerically grow under the *weak boundary problem*, the *first arrival* probability of that label is increased without regard to the whole relation between a pixel and seeds. These limitations explain why RW still suffers from the two problems: the *weak boundary problem* and the *texture problem*. Most recently, the segmentation approach defined by an l_∞ norm was proposed in [7]. Since RW with constraints was used as a regularization method for yielding a unique solution, this approach still has the limitations of RW.

Most previous supervised image segmentation algorithms focus on the inter-label discrimination, not finding the generative model for each label. Although they tried to solve the *weak boundary problem* and the *texture problem*, these two problems in natural images are still the most challenging issues in image segmentation. In this paper, we propose a new generative image segmentation algorithm based on the Random Walks with Restart (RWR) [8] that can solve the *weak boundary problem* and *texture problem* effectively. The key contributions of our proposed algorithm are as follows.

1. We introduce a generative model for image segmentation. From basic decision theory [9], it is known that generative methods are better inference algorithms. In contrast to most existing models which focus on the inter-label discrimination, we address the problem of finding the generative model for each label like [10]. For example, we can consider just one-label segmentation problem as shown in Fig. 1. Our model can produce the segmentation result with an optimal threshold level as shown in Fig. 1(c). This is possible since the likelihood probability can be generated using our generative model as depicted in Fig. 1(b). Since the generative model of each label is constructed independently, it is also possible to add a new label without altering the models of previous labels.

(a) Original (b) Likelihood (c) Resulting segmentation

Fig. 1. An example of the generative segmentation with just one label. Given the seeds with green initial label in (a), the likelihood in (b) is computed using our generative algorithm. The range of this probability is $[0,\ 2.8225 \times 10^{-4}]$. (c) is the resulting segmentation with a threshold level $\tau = 1.5524 \times 10^{-5}$. The foreground label is assigned to the pixels with probability above the threshold τ.

2. We design a generative image segmentation approach using the *steady-state* probability of RWR as a part of the likelihood term. Since the likelihood of a pixel is defined as the average of all the *steady-state* probabilities between that pixel and the seeds with same label, our algorithm can reduce dependence on the number of seeds under the *weak boundary problem*. RWR, similar to graph-based semi-supervised learning [11], is a very successful technique for defining the relevance relation between two nodes in graph mining [8][12][13][14]. It has good performance on many other applications: Cross-model correlation discovery [8], Center-piece subgraph discovery [13], Content based image retrieval [12], Neighborhood formulation [14] and etc. Since this *steady-state* probability of RWR considers the whole relationship between two pixels, it naturally reflects the effects of texture.
3. Under two challenging problems: the *weak boundary problem* and the *texture problem*, our algorithm produces very good segmentation results on synthetic and natural images. It has better performance than RW as well as GC.

The paper is organized as follows. In Section 2, we introduce our proposed generative image segmentation algorithm and explain that algorithm in detail. The experimental results are shown in Section 3. Finally, we discuss our approach and give conclusions in Section 4.

2 Generative image segmentation

Let us consider the image segmentation as a labeling problem in which each pixel $x_i \in \mathbf{X} = \{x_1, .., x_N\}$ is to be assigned one label $l_k \in \mathbf{L} = \{l_1, .., l_K\}$. From basic decision theory [9], we know that the most complete characterization of the solution is expressed in terms of the set of posterior probabilities $p(l_k|x_i)$. Once we know these probabilities, it is straightforward to assign x_i the label having the largest probability. In a generative approach, we model the joint distribution $p(l_k, x_i)$ of pixels and labels. This can be done by computing the

label prior probability $p(l_k)$ and the pixel likelihood $p(x_i|l_k)$ separately. The required posterior probabilities are obtained using Bayesian rules:

$$p(l_k|x_i) = \frac{p(x_i|l_k)p(l_k)}{\sum_{n=1}^{K} p(x|l_n)p(l_n)}, \tag{1}$$

where the sum in the denominator is taken over all labels.

Let $\mathbf{X}^{l_k} = \{x_1^{l_k}, ..., x_{M_k}^{l_k}\}$ ($\mathbf{X}^{l_k} \subset \mathbf{X}$) be a set of the M_k seeds with label l_k. Then the likelihood $p(x_i|l_k)$ can be obtained by

$$\begin{aligned} p(x_i|l_k) &= \tfrac{1}{Z} \textstyle\sum_{m=1}^{M_k} p(x_i|x_m^{l_k}, l_k)p(x_m^{l_k}|l_k) \\ &= \tfrac{1}{Z \times M_k} \textstyle\sum_{m=1}^{M_k} p(x_i|x_m^{l_k}, l_k), \end{aligned} \tag{2}$$

where Z is a normalizing constant. Each pixel likelihood is modeled by a mixture of distribution $p(x_i|x_m^{l_k}, l_k)$ from each seed $x_m^{l_k}$ which has a seed distribution $p(x_m^{l_k}|l_k)$. The pixel distribution $p(x_i|x_m^{l_k}, l_k)$ indicates the relevance score between a pixel x_i and a seed $x_m^{l_k}$. In this work, we propose to use the *steady-state* probability defined by RWR [8]. Compared with traditional graph distances (such as shortest path, maximum flow), this *steady-state* probability can capture the whole relationship between two pixels. The seed distribution $p(x_m^{l_k}|l_k)$ is defined by a uniform distribution, $1/M_k$. Since the likelihood $p(x_i|l_k)$ is computed by the average of the pixel distributions of all the seeds with the label l_k, our method is less dependence on the number of seeds.

Now, we briefly describe the process of our image segmentation algorithm. First, we construct a weighted graph in an image. Then, we define $p(x_i|x_m^{l_k}, l_k)$ as the *steady-state* probability that a random walker starting at a seed $x_m^{l_k}$ stays at a pixel x_i in this graph. After computing this *steady-state* probability using RWR, we can estimate the likelihood $p(x_i|l_k)$ in (2) and, finally assign the label with maximum posterior probability in (1) to each pixel.

2.1 Graph Model

Given an image I, let us construct an undirected graph $G = (V, E)$ with nodes $v \in V$, and edges $e \in E$. Each node v_i in V uniquely identifies an image pixel x_i. The edges E between two nodes are determined by the neighborhood system. The weight $w_{ij} \in W$ is assigned to the edge $e_{ij} \in E$ spanning between the nodes $v_i, v_j \in V$. It measures the likelihood that two neighboring nodes v_i, v_j have the same label.

The weights encode image color changes used in many graph based segmentation algorithms [15][4][6]. In this work, a weight w_{ij} is defined as the typical Gaussian weighting function given by

$$w_{ij} = exp\left(-\frac{\|g_i - g_j\|^2}{\sigma}\right), \tag{3}$$

where g_i and g_j indicate the image colors at two nodes v_i and v_j in *Lab* color space. It provides us with a numerical measure, a number between 0 and 1, for

the similarity between a pair of pixels. The gaussian function has the nature of geodesic distance. For example, the multiplication between two weights w_{ij}, w_{jk}, $w_{ij}w_{jk} = exp\left(-\frac{\|g_i-g_j\|^2+\|g_j-g_k\|^2}{\sigma}\right)$, can measure the similarity between the nodes v_i, v_k. This property fits in well with our algorithm. Therefore, we choose this Gaussian function for the edge weights.

2.2 Learning

Suppose a random walker starts from a m-th seed pixel $x_m^{l_k}$ of label l_k in this graph G. The random walker iteratively transmits to its neighborhood with the probability that is proportional to the edge weight between them. Also at each step, it has a *restarting* probability c to return to the seed $x_m^{l_k}$. After convergence, we obtain the *steady-state* probability $r_{im}^{l_k}$ that the random walker will finally stay at a pixel x_i. In this work, we use this *steady-state* probability $r_{im}^{l_k}$ as the distribution $p(x_i|x_m^{l_k}, l_k)$ in (2) such as

$$p(x_i|x_m^{l_k}, l_k) \approx r_{im}^{l_k}. \tag{4}$$

By denoting $r_{im}^{l_k}, i = 1, ..., N$ in terms of an N-dimensional vector $\boldsymbol{r}_m^{l_k} = [r_{im}^{l_k}]_{N\times 1}$ and defining an adjacency matrix $\mathbf{W} = [w_{ij}]_{N\times N}$ using (3), RWR can be formulated as follows [8].

$$\begin{aligned} \boldsymbol{r}_m^{l_k} &= (1-c)\mathbf{P}\boldsymbol{r}_m^{l_k} + c\boldsymbol{b}_m^{l_k} \\ &= c(\mathbf{I}-(1-c)\mathbf{P})^{-1}\boldsymbol{b}_m^{l_k} \\ &= \mathbf{Q}\boldsymbol{b}_m^{l_k}, \end{aligned} \tag{5}$$

where $\boldsymbol{b}_m^{l_k} = [b_i]_{N\times 1}$ is the N-dimensional indicating vector with $b_i = 1$ if $x_i = x_m^{l_k}$ and $b_i = 0$ otherwise, and the transition matrix $\mathbf{P} = [p_{ij}]_{N\times N}$ is the adjacency matrix $\mathbf{W}$ row-normalized:

$$\mathbf{P} = \mathbf{D}^{-1} \times \mathbf{W}, \tag{6}$$

where $\mathbf{D} = diag(D_1, ..., D_N)$, $D_i = \sum_{j=1}^{N} w_{ij}$. If these *steady-state* probabilities $\boldsymbol{r}_m^{l_k}$ are inserted into (2) by our definition, the likelihoods $p(x_i|l_k)$ $(i = 1, ..., N)$ are achieved such as:

$$[p(x_i|l_k)]_{N\times 1} = \frac{1}{Z \times M_k}\mathbf{Q}\tilde{\boldsymbol{b}}^{l_k}, \tag{7}$$

where $\tilde{\boldsymbol{b}}^{l_k} = [\tilde{b}_i]_{N\times 1}$ is the N-dimensional vector with $\tilde{b}_i = 1$ if $x_i \in \mathbf{X}^{l_k}$ and $\tilde{b}_i = 0$ otherwise.

In a RWR view, $\mathbf{Q} = [q_{ij}]_{N\times N}$ is used for computing the affinity score between two pixels. In other words, q_{ij} implies the likelihood that q_i has the same label which was assigned to q_j. It can be reformulated as follows:

$$\begin{aligned} \mathbf{Q} &= c(\mathbf{I}-(1-c)\mathbf{P})^{-1} \\ &= c\sum_{t=0}^{\infty}(1-c)^t\mathbf{P}^t. \end{aligned} \tag{8}$$

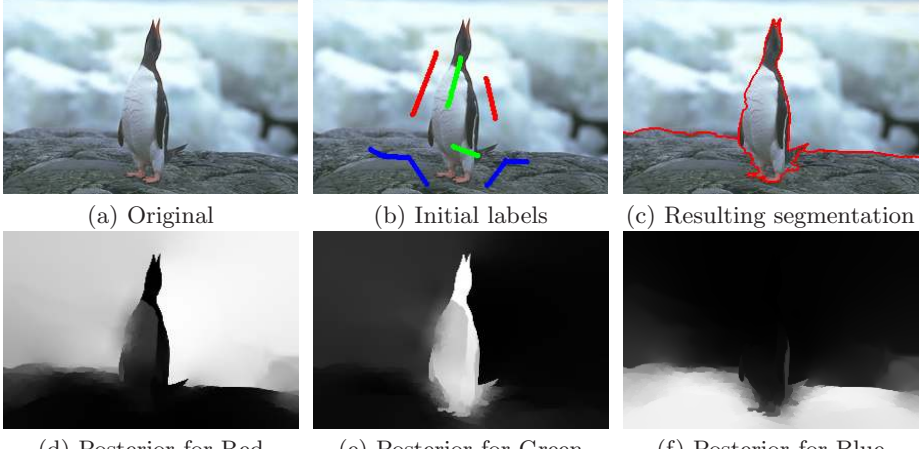
$\mathbf{Q}$ is defined as the weighted sum of all matrices $\mathbf{P}^t, t = 0, ..., \infty$. Note that $\mathbf{P}^t$ is the t-th order transition matrix, whose elements p_{ij}^t can be interpreted as the total probability for a random walker that begins at v_j to end up at v_i after t iterations, considering all possible paths between two pixels. By varying the number of iterations t, we explicitly explore relationship at different scales in the image, and as t increases, we expect to find coarser structure. Therefore, RWR gives the effects of texture by considering *all paths* between two pixels at all scales (*any* iteration number ($t = 0, ..., \infty$)) in the image. Since close-by pixels are likely to have high similarity value, as t increases, $\mathbf{P}^t$ has lower weight, $(1-c)^t$ ($0 < c < 1$). The resulting matrix $\mathbf{Q}$ can be solved by using a linear method of matrix inversion. Although it requires more memory space, fast computation is possible if the matrix is sparse. Since the small pixel neighborhood system is used in this paper, normalized matrix $\mathbf{P}$ is highly sparse. Therefore, $\mathbf{Q}$ can be calculated fast. However, if the number of nearest neighbors is chosen to be a fixed large number, the complexity of matrix inversion is very high. In this case, some approximation methods such as Fast Random Walk with Restart method [16] can be used.

2.3 Segmentation

Assume that the prior probability $p(l_k)$ in (1) is uniform. Using the likelihood $p(x_i|l_k)$ in (7), the decision rule of each pixel x_i for image segmentation is as follows:

$$R_i = \arg\max_{l_k} p(l_k|x_i) = \arg\max_{l_k} p(x_i|l_k). \tag{9}$$

By assigning the label R_i to each pixel x_i, the segmentation is obtained.



(a) Original (b) Initial labels (c) Resulting segmentation

(d) Posterior for Red (e) Posterior for Green (f) Posterior for Blue

Fig. 2. Overview of our proposed segmentation algorithm. Given seeds (Red, Green and Blue) in (b), the posterior probabilities (d),(e) and (f) are obtained by computing (1) for the three labels, respectively. Segmentation result (c) is obtained by assigning each pixel the label having maximum posterior probability.

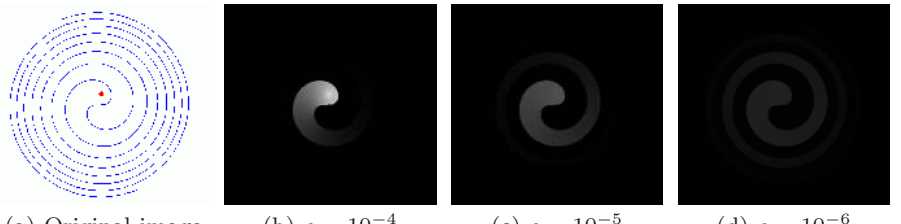
Fig. 2 shows the overall process of our algorithm from the seeds to the calculation of each label posterior probability $p(l_k|x_i)$ and the resulting segmentation. It starts with three initial seed labels: Red, Green, and Blue as shown in Fig. 2(b). After computing the likelihood for each label, we generate the posterior probabilities as shown in Fig. 2(d),(e) and (f). By a decision rule (9), each pixel is assigned the label that has the maximum probability. Finally, we obtain the segmentation results in Fig. 2(c), where the object boundary is drawn in red color overlaid on the original image.

3 Experimental Results

Our algorithm has two parameters: a color variance σ and a *restarting* probability c. The σ is used in all graph-based segmentation algorithms. In this work, this parameter is fixed with the same value for all the segmentation algorithms we tested. However, the *restarting* probability c is only used in our algorithm, for computing the *steady-state* probability of RWR. In this section, we first analyze the effect of the *restarting* probability c in image segmentation. And then, we compare the performance of our algorithm with the state of the art methods including GC [4] and RW [6] on several synthetic and natural images.

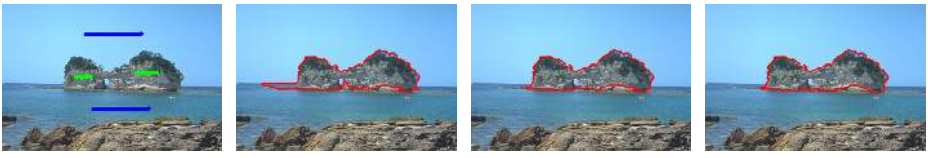
3.1 Parameter Setting

RWR needs one parameter; the *restarting* probability c in (5). According to this parameter, the range of propagation of a random walker from the starting node is varied as shown in Fig 3. If c is decreased, the probability that a random walker travels over a larger area is increased. This means that by varying c, we can control the extent of the label information of a seed in different scales in the image. Figure 4 shows another example of the segmentation with respect to the variation of the *restarting* probability c in a natural image. According to the *restarting* probability c, the segmentation results are changed. Therefore, it is important to find a proper probability c according to the quantity (or quality)



(a) Original image (b) $c = 10^{-4}$ (c) $c = 10^{-5}$ (d) $c = 10^{-6}$

Fig. 3. An example of the variability of the *steady-state* probabilities $\boldsymbol{r}$ according to the *restarting* probability c. (a) is an image whose size is 328×310. It has the seeds with just one label (red pixels) at the center. (b),(c) and (d) show the variation of $\boldsymbol{r}$ in accordance with the decrease of the *restarting* probability c in a 4-connected pixel neighborhood system. The display range of $\boldsymbol{r}$ is $[0, 0.0006]$.



(a) Original image | a_o=0.826165 (b) $c = 10^{-3}$ | a_o=0.850946 (c) $c = 10^{-4}$ | a_o=0.877542 (d) $c = 10^{-5}$

Fig. 4. An example of the segmentation with respect to the variation of the *restarting* probability c in a natural image. For comparison, GC has a_o=0.789613 and RW has a_o=0.781148. The accurate rate a_o is computed by (10).

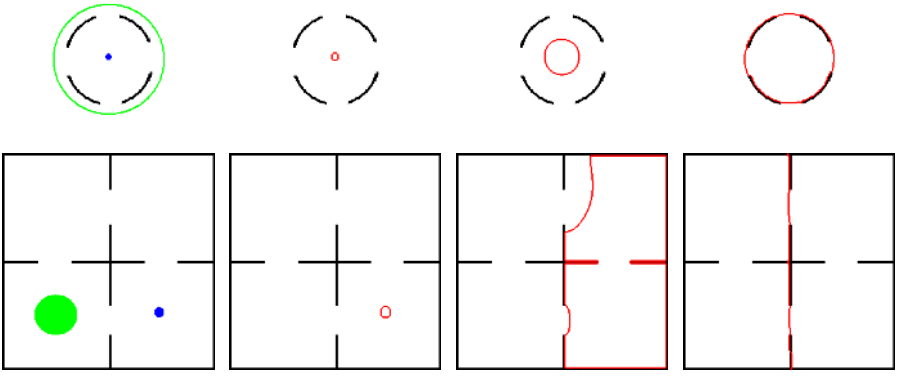
of the seeds and the image size. In this work, c was chosen empirically, and we set $c = 4 \times 10^{-4}$ for all the natural test images. The 8-connected neighborhoods were used as the pixel neighborhood system.

3.2 Segmentation Results

We begin by analyzing the performance under two difficult problems: *weak boundary problem* and *texture problem*. We then compare the segmentation results obtained by the three algorithms on natural images and provide quantitative comparisons.

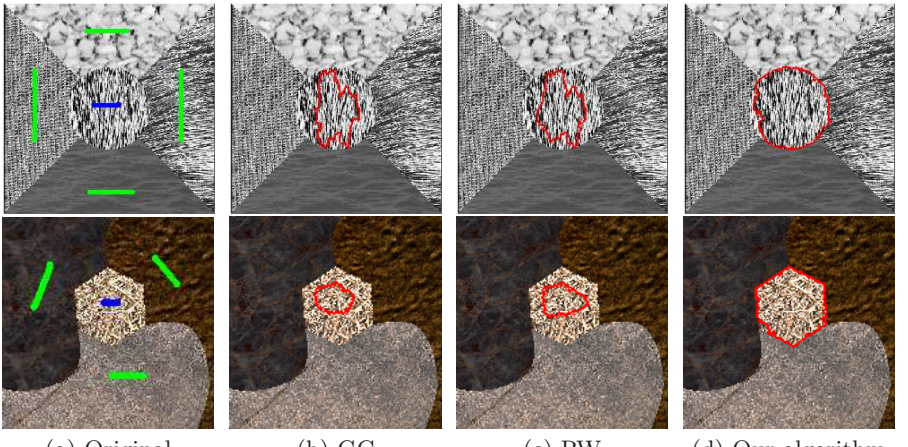
Weak boundary problem. The weak boundary problem is to find the weak boundaries when they are parts of a consistent boundary. In [6], RW shows better segmentation result than GC in low contrast with small number of seeds. Although GC and RW are capable of finding weak boundaries, our algorithm gives more intuitive outputs. In Fig. 5, our algorithm is compared with GC and RW in the *weak boundary problem*. We used two synthetic examples: circle and 3×3 grid with four sections erased. Given the seeds (*green* and *blue*) in Fig. 5(a), the segmentations were obtained in Fig. 5(b)-(d). Fig. 5(b) shows clearly that GC has *small cut* problem. In Fig. 5(c), we can confirm that the segmentations of RW is substantially affected by the difference between the numbers of Green and Blue seeds. Namely, RW is sensitive to the number of seeds. In contrast, Fig. 5(d) shows that our algorithm is less dependent on the number of seeds and produces better segmentations, because the likelihood is computed by the average of the relevance scores of all the seeds.

Texture problem. In GC and RW, it is hard to separate the textured region without considering higher-order connections, because they deal with the minimum cut criterion and the *fisrt arrival* probability, respectively. Since these two algorithms do not consider the information of seeds that are inside the prelabeled regions, it is not easy for them to take into account the effects of texture. On the other hand, our algorithm can reflects the texture information by using the *steady-state* probability of RWR, because RWR considers all possible paths between two nodes in a small neighborhood system. In spite of the use of a small neighborhood system, it captures the textural structure well and obtains



(a) Original (b) GC (c) RW (d) Our algorithm

Fig. 5. Comparison of our algorithm with GC and RW for finding weak boundaries. Given two labels (Green and Blue) and original images in (a) (Top: image created with a black circle with four erased sections. Bottom: image created with a 3×3 black grid with four erased sections.), (b),(c) and (d) are the segmentation results of GC, RW and our algorithm ($c = 4 \times 10^{-4}$), respectively.



(a) Original (b) GC (c) RW (d) Our algorithm

Fig. 6. Comparison of our algorithm ($c = 10^{-6}$) with GC and RW on synthetic textured images

object details. In Fig. 6, we used the synthetic images that consist of four or five different kinds of textures. This is the texture segmentation problem that one texture is extracted among them. The segmentation results in Fig. 6(d) show that our algorithm produced more reliable texture segmentations on these synthetic textured images than GC and RW.

Quantitative comparisons. The previous two situations often arise in natural images. Now, we compare the segmentations obtained from GC, RW, and our

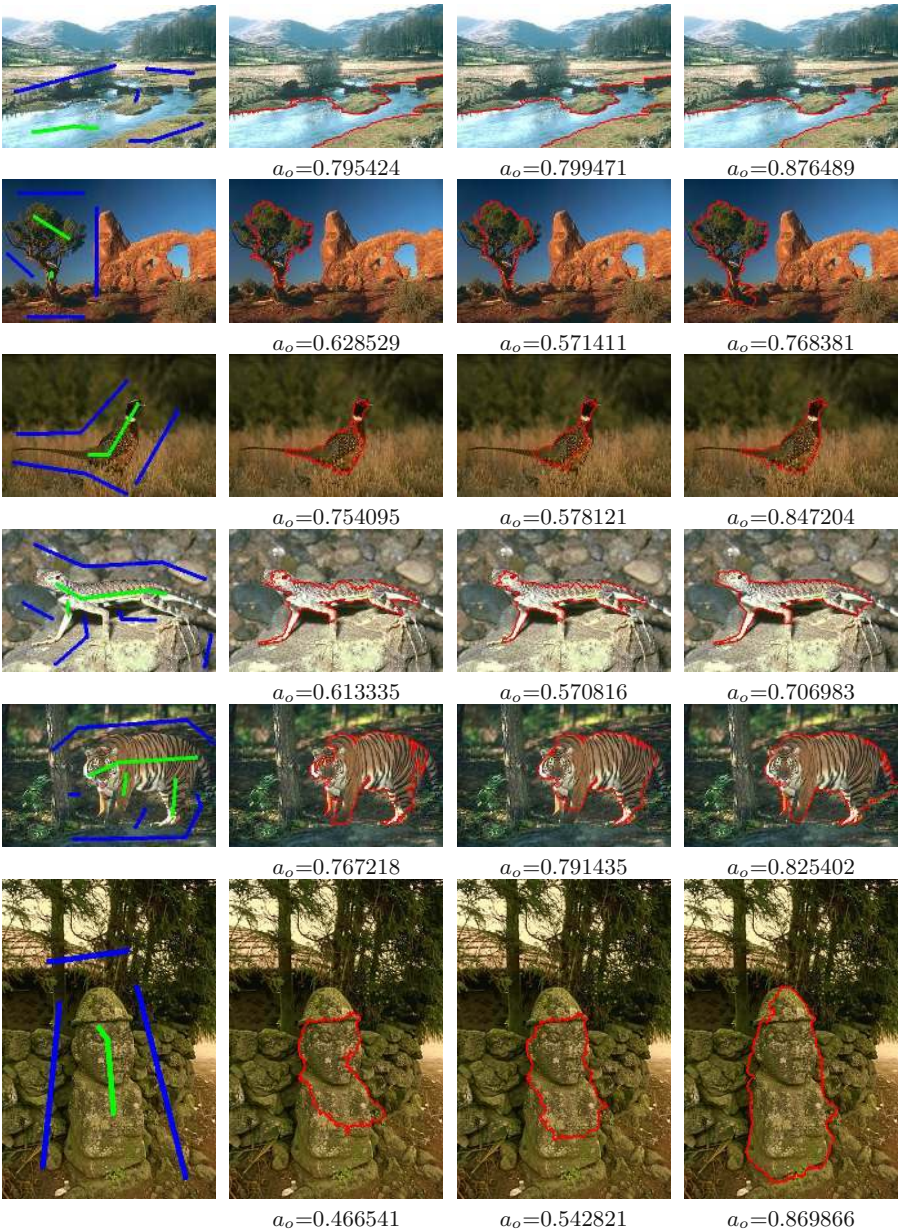


Fig. 7. Comparison of our algorithm with RW and GC on natural images. (b),(c) and (d) are the segmentation results of GC, RW and our algorithm ($(c = 4 \times 10^{-4})$), respectively. a_o is the accurate rate in (10).

algorithm on natural images. We utilized a dataset of natural images where human subjects provide foregrond/background label: ground truth (the Berkeley Segmentation Dataset [17]). For quantitative comparisons, the similarity between the segmentation result and the preset ground truth was measured using a normalized overlap a_o [7]:

$$a_o = \frac{|R \cap G|}{|R \cup G|}, \tag{10}$$

where R is the set of pixels assigned as the foreground from the segmentation result and G is that from the ground truth. In this work, it was used as the *accuracy* measure of the image segmentation. For this experiments, we chose the natural images with highly textured (cluttered) regions or with similar color distributions between the foregrounds and backgrounds. In Fig. 7, the segmentations were produced from the three different algorithms on these natural images. Compared with the segmentations from GC and RW in Fig. 7, our algorithm has better segmentations qualitatively and quantitatively. The quantitative comparison confirms the relevance and accuracy of our algorithm.

4 Conclusion

This paper presents a novel generative image segmentation model in the Bayesian Framework. More importantly, we provide a new interpretation of RWR for image segmentation. Although RW [6] is also based on the Random Walks concept, our work is conceptually different from RW, and produces significant improvement in performance as shown in the experiments.

The key differences between RW and our work are "First arrival probability vs. Average probability". In [6], the score between a pixel and each label is defined by the *first arrival* probability that a random walker starting at a pixel reaches to a seed. On the other hand, in our work it is defined as the average probability that a random walker starting at one of the seeds stays at a pixel.

Our approach has several advantages for image segmentation. First, owing to the generative segmentation model with RWR, it can obtain segmentations with just single label. Second, it is less dependent on the number of seeds, because the likelihood is computed by the average of the relevance scores of all the seeds. Finally, it gives *qualitatively* and *quantatively* better segmentations on natural images. Generally, large neighborhood system is needed for obtaining object details, because it captures image structure well. Since it gives high computations, many efficient methods, like multi-scale approach, have been proposed. Our method is an alternative solution, since RWR considers all possible paths between two nodes in small neighborhood system.

For the computation of RWR, in this work, the *restarting* probability c, was chosen empirically. However, it is not optimal for every image. If we can control it well, better segmentation results will be obtained. Thus, our future work will include the automatic selection of the optimal value of this parameter.

Acknowledgement

This research was supported in part by the Defense Acquisition Program Administration and Agency for Defense Development, Korea, through the Image Information Research Center under the contract UD070007AD, and in part by the MKE (Ministry of Knowledge Economy), Korea under the ITRC (Information Technolgy Research Center) Support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2008-C1090-0801-0018).

References

1. Mortensen, E.N., Barrett, W.A.: Interactive segmentation with intelligent scissors. Graphical Models in Image Process. 60(5), 349–384 (1998)
2. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. IJCV V1(4), 321–331 (1988)
3. Sethian, J.A.: Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge University Press, Cambridge (1999)
4. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient n-d image segmentation. IJCV 70(2), 109–131 (2006)
5. Bai, X., Sapiro, G.: A geodesic framework for fast interactive image and video segmentation and matting. In: Proc. ICCV 2007, pp. 1–8 (2007)
6. Grady, L.: Random walks for image segmentation. PAMI 28(11), 1768–1783 (2006)
7. Sinop, A.K., Grady, L.: A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. In: Proc. ICCV (2007)
8. Pan, J.Y., Yang, H.J., Faloutsos, C., Duygulu, P.: Automatic multimedia cross-modal correlation discovery. In: KDD 2004, pp. 653–658 (2004)
9. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (1995)
10. Grady, L., Schwartz, E.L.: Isoperimetric graph partitioning for image segmentation. PAMI 28(3), 469–475 (2006)
11. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Scholkopf, B.: Learning with local and global consistency. In: NIPS (2003)
12. He, J., Li, M., Zhang, H.J., Tong, H., Zhang, C.: Manifold-ranking based image retrieval. In: ACM Multimedia, pp. 9–16 (2004)
13. Tong, H., Faloutsos, C.: Center-piece subgraphs: problem definition and fast solutions. In: KDD 2006, pp. 404–413 (2006)
14. Sun, J., Qu, H., Chakrabarti, D., Faloutsos, C.: Neighborhood formation and anomaly detection in bipartite graphs. In: ICDM 2005, pp. 418–425 (2005)
15. Shi, J., Malik, J.: Normalized cuts and image segmentation. PAMI 22(8), 888–905 (2000)
16. Tong, H., Faloutsos, C., Pan, J.Y.: Fast random walk with restart and its applications. In: ICDM 2006, pp. 613–622 (2006)
17. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. ICCV, vol. 2, pp. 416–423 (2001)