Interactive Segmentation with Seed Expansion

Gwangmo Song, Heesoo Myeong, and Kyoung Mu Lee
Department of Electrical and Computer Engineering, ASRI, Seoul National University, Seoul, Korea
E-mail: kfsgm@snu.ac.kr, heesoo.myeong@gmail.com, kyoungmu@snu.ac.kr

***Abstract*—In this paper, we address the problem of interactive image segmentation which segments an image based on user-supplied scribbles. For this purpose, we propose a novel framework that provides consistent performance robust to the location of input seeds. Most of the existing methods, especially random walk-based approaches, strongly depend on initial seed positions, which differ from one user to another. To overcome this drawback, the proposed algorithm incorporates the seed expansion step, in which robust seed information is secured and improved. We also address the computation issue using a coarse-to-fine random walk technique. We evaluate our algorithm using challenging datasets, such as Grabcut, PASCAL VOC and Alpha matting datasets. Our algorithm produces a more accurate segmentation results than the existing methods.**

Fig. 1. Example of interactive image segmentation. Image and seeds (left) and segmentation result (right). Red seeds represent the foreground and orange seeds represent the background. Output segmentation boundary is shown as cyan contour.

I. Introduction

Image segmentation is one of the biggest problems in computer vision. Other than the application itself, image segmentation plays a significant role as a base technique in other computer vision problems. Despite the development of various unsupervised image segmentation methods, local appearance ambiguities intrinsically limit the performance. Hence, for better practicability, semi-supervised image segmentation methods, called interactive image segmentation techniques are gaining popularity in segmenting the object with less human effort [9], [10], [11]. In this paper, we address problems that arise from interactive image segmentation.

The problem of interactive image segmentation is the typical use of small foreground/background seeds from users to extract the foreground region. In some studies, interactive image segmentation problem is formulated using a graph cut framework [1], [2], [3], whereas others, employ the random walk algorithm [4], [5], [6]. However, these studies had not solved the technical problems arising from interactive image segmentation, and did not exhibit satisfactory results for practical applications. The recent random walk with restart-based algorithm [6], shows a promising performance, but much improvement can still be made.

One of the problems of the random walk-based algorithm is that its performance is largely affected by the configuration of initial seeds. That is, because of the nature of the algorithm, whenever the user changes the seed, the result also changes. Moreover, when the user provides foreground and background seeds, the algorithm forms a boundary around the middle of foreground and background seeds. Thus, the algorithm is sensitive to the distance from the seed. In this paper, we aim to improve the performance of the algorithm by introducing solutions to the problems.

To solve the problem, we update the initial seeds instead of using them as fixed seeds. The configuration of the original seeds does not generally convey enough information because we assume that minimal amount of human effort is required. Thus, if used alone, accurate results cannot be obtained. Therefore, if we modify seeds progressively by boosting them to fulfill the intention of a user, we can expect improved segmentation results.

Moreover, one of the drawbacks of the baseline random walk algorithm is slow speed. Speed cannot be overlooked because it plays a crucial role in real applications. In this paper, we propose a coarse-to-fine random walk approach to significantly enhance the speed.

II. Interactive Segmentation with Random Walk with Restart (RWR)

First, let us begin by briefly explaining the random walk with restart (RWR) method for interactive segmentation which is our baseline algorithm [6]. The method calculates the posterior probability of foreground/background labels on a graph, which represents a given image. The RWR algorithm is used to obtain the likelihood of the labels. Random walkers start from the given foreground/background seed pixels, traverse the graph, and explore the structure of the graph. The probability of the movement depends on the edge weight among the pixels. This method is similar to the interactive segmentation method that is based on the random walk (RW) [4]. In the RWR algorithm, the restarting probability is added. According to the restarting probability c, a random walker returns to its user-supplied seed pixels at a constant probability. In other words, a random walker makes a transition to its closest pixel at a probability of $1 - c$ and then restarts from its seed pixels at a probability of c. Finally, repeated transitions and

978-616-361-823-8

restarts establish a steady-state probability, which is regarded as a likelihood of each label. This algorithm is formulated mathematically as,

$$\mathbf{r}_m^l = (1-c)\mathbf{P}\mathbf{r}_m^l + c\mathbf{b}_m^l, \tag{1}$$

where $\mathbf{r}$ represents the steady-state probability for the m-th seed of each label l which indicates either the foreground or the background, c stands for the restarting probability, transition matrix $\mathbf{P}$ contains the edge weight, and $\mathbf{b}$, which is called indicating vector, contains the information on the locations of the seeds.

To apply RWR to image segmentation, an image should be represented as a graph, and then the image segmentation can generally be modeled by an undirected graph $G = (V, E)$ with nodes $v_i \in V$ and edges $e_{ij} \in E$. Each node v_i corresponds to a pixel x_i of a given image $X = \{x_1, ..., x_N\}$. The edge weight w_{ij} represents the strength of e_{ij} or how two neighboring nodes are similar, which is given by $w_{ij} = exp(-|g(x_i) - g(x_j)|^2/\sigma)$ using the simple Gaussian weighting function. The function $g(\cdot)$ represents the image colors of each pixel in Lab color space. Note that the component of $\mathbf{r}_m^l$ that corresponds to pixel x_i can be considered as the likelihood of a single seed at pixel x_i, $p(x_i|x_m^l, l)$. We can calculate the total likelihood of a label l at a pixel x_i by

$$p(x_i|l) = \frac{1}{Z \times |M_l|} \sum_{m=1}^{M_l} p(x_i|x_m^l, l), \tag{2}$$

where Z is a normalizing constant and M_l is the total number of seeds of label l. The RWR algorithm is not affected by the number of seed pixel because of the normalizing factor. However no stabilizer was used for the change in position of the seed pixel. From this likelihood, we can obtain the posterior probability of each label,

$$p(l|x_i) = \frac{p(x_i|l)p(l)}{\sum_l p(x_i|l)p(l)}. \tag{3}$$

We assume the prior probability $p(l)$ as uniform distribution. From this posterior probability, foreground/background label is directly assigned to each pixel x_i.

III. The Proposed Method

A. Pyramid Structure

An accurate result of the interactive image segmentation is critical as well as the speed of the process. In the RWR segmentation, we can calculate the steady-state probability by either the iterative method known as power iteration or the simple inversion of the matrix.

$$\begin{aligned}\mathbf{r}_m^l &= (1-c)\mathbf{P}\mathbf{r}_m^l + c\mathbf{b}_m^l \\ &= c(I - (1-c)\mathbf{P})^{-1}\mathbf{b}_m^l. \end{aligned} \tag{4}$$

The computation result is significant when power iteration is used because of its large repetition number. The matrix inversion also requires large computations. To reduce the overall computational complexity, we construct an image pyramid and adopt a coarse-to-fine strategy. A three-level Gaussian pyramid is used in the present study. First, the algorithm is applied to the coarsest level with less computational cost, and then the solution is projected onto the next level by a refinement step. In this strategy, the final result is obtained at the finest level. Using this coarse-to-fine strategy, we can considerably enhance both the convergence rate and the robustness to local appearance variations by considering global context in the image.

Fig. 2. Failure case of the RWR algorithm. The image at the left represents an original image and the corresponding scribbles from the user. The image at the right is the result of the interactive segmentation using RWR algorithm. Red contour represents the obtained boundary. The segmentation boundary occurs in between the foreground and the background scribbles.

B. Seed Expansion

One of the problems of the RWR algorithm is that the segmentation boundary highly depends on the positions of both the foreground and the background seeds, as shown in Figure 2. This dependence usually causes two problems: spreading and shrinking problems. The spreading problem causes the segmentation boundary to either be open or expand beyond the image boundary, whereas the shrinking problem prevent the full expansion of the segment region. These problems are the typical drawbacks of the random walk-based methods. In this paper, we modify the seed information instead of the random walk algorithm itself. Therefore, to solve these problems, we suggest adjusting the region of the seeds to place the boundary of the object in the middle of the modified foreground and the background seeds.

To expand the seed, an image is first broken into superpixels by watershed oversegmentation [12]. Subsequently, if any of the superpixel contains a seed pixel, we assign the label of the seed pixel to all the pixels in the superpixel. Consequently, the seed regions are expanded.

Note that the seeds must be extended to become close to the object's boundary. We bring the superpixel seeds to the coarsest level of the pyramidal structure and obtain the likelihoods of the labels by a few power iterations. Larger

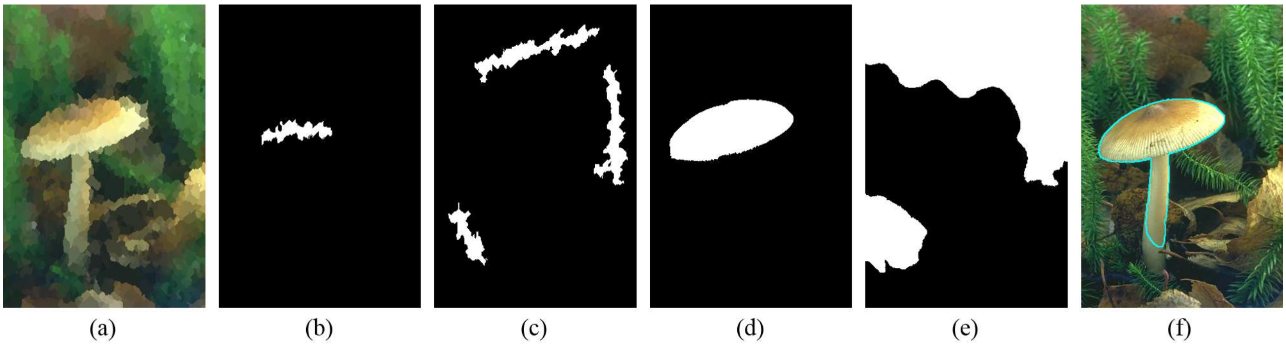

Fig. 3. Procedure of seed expansion. (a) Obtained watershed superpixels of the original image. Each superpixel has a similar color. (b), (c) The set of the superpixels contains foreground or background scribbles. (d), (e) The expanded seeds by RWR power iteration and thresholding. (f) The segmentation result of the proposed algorithm.

and reliable seeds are then obtained by the thresholds of the likelihood values with a conservative threshold value. Segmentation result is achieved using these finally expanded seeds at the coarsest level and then recursively projected and refined at the finer levels, which result in a final segmentation boundary at the original scale. The proposed seed expansion method produces robust results to the changes of the initial seed, as configured by the users.

IV. Experimental Results

A. Dataset

To evaluate the algorithm, an experiment is conducted using a dataset that is generally considered difficult to solve. Gulshan *et al.* [7] suggested an intricate dataset by collecting contemporary datasets, comprising a total of 49 sheets of Grabcut dataset images [3], 99 sheets of PASCAL VOC'09 segmentation challenge dataset [13] and 3 sheets of alpha-matting dataset images [14]. Each dataset contains an original image and a seed image, as well as a ground truth image. Each seed image consists of 1 line of foreground seed and 3 lines of background seeds. Numerous current interactive image segmentation studies had conducted experiments on datasets using seeds given as tight tri-map. However, as a practical approach, our experiment is conducted using a complicated dataset.

B. Qualitative & quantitative results

In these experiments, the proposed algorithm is compared with two state-of-the art algorithms; GSCseq algorithm [7] and RWR algorithm [6]. For a quantitative evaluation, the dice coefficient is used to measure the similarity between the result of the segmentation and the ground truth.

TABLE I
AVERAGE ACCURACY OF THE SEGMENTATION ALGORITHMS

Method	Accuracy
GSCseq	70.7 %
RWR	70.5 %
Our Algorithm	**76.1 %**

Fig. 4. Effect of the spreading and shrinking problem. The left images are the results of the RWR algorithm. Top-left image shows that the RWR algorithm has the shrinking problem and the bottom-left image shows the spreading problem. In our algorithm (right images), the effect of these problem are lessened.

$$accuracy = \frac{2|R \cap G|}{|R| + |G|} \times 100(\%), \qquad (5)$$

where R represents a segmented region and G represents a ground truth region. Average accuracies of each algorithm are shown in Table I. The results of GSCSeq and RWR algorithm are easily distinguished as similar, whereas our proposed algorithm produces the best result.

Figure 4 shows some examples of spreading and shrinking problems of the RWR algorithm. The top-left and bottom-left images both exhibit the shrunk and the spread results of RWR. The images in the right column are the corresponding results of the proposed algorithm, where the spreading and shrinking problems are clearly resolved. Figure 5 shows the qualitative comparison of the results of the three algorithms. Apparently, our algorithm shows the best results compared with those of the other two.

The average running time of the proposed algorithm is 0.682s, which is much faster than RWR that has average

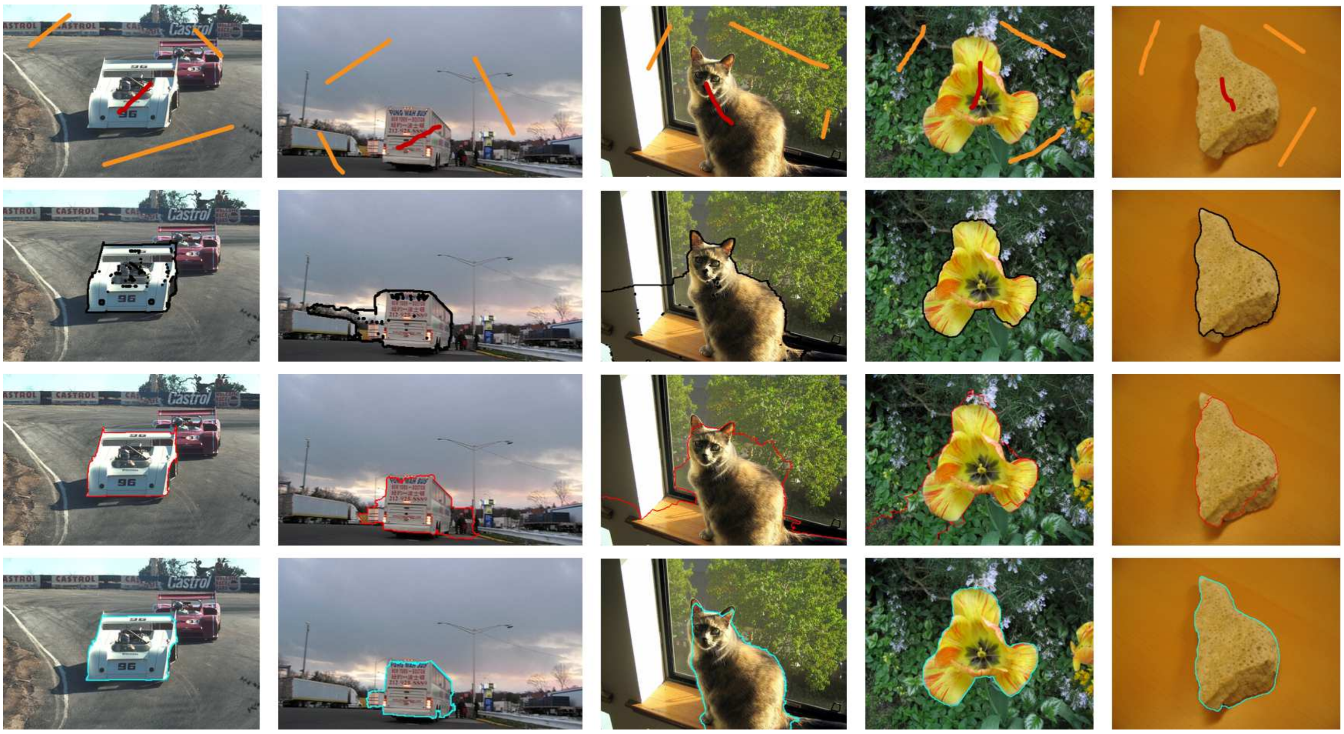

Fig. 5. Segmentation results. The first row contains the original images and seeds. Second row shows the segmentation results of the GSCseq algorithm. Third row shows the results of the RWR algorithm. The last row shows the results of our proposed algorithm.

runtime of 2.331s on a 3.3GHz Intel quadcore i5 CPU and 8GB RAM.

V. Conclusion

We have proposed a new interactive image segmentation framework by RWR via seed expansion. The proposed algorithm produces highly stable and accurate segmentation results by aggregating reliable seeds and also allows fast convergence by the coarse-to-fine strategy. The experimental results demonstrate that the proposed algorithm has superior performance compared with the existing algorithms.

Acknowledgements

This work was partly supported by the ICT R&D program of MSIP/IITP, Korea [14-824-09-006, Novel Computer Vision and Machine Learning Technology with the Ability to Predict and Forecast].

References

[1] Y. Boykov and M.-P. Jolly, “Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images,” In *Proc. International Conference on Computer Vision (ICCV)*, 2001.

[2] V. S. Lempitsky, P. Kohli, C. Rother, and T. Sharp, “Image segmentation with a bounding box prior,” In *Proc. International Conference on Computer Vision (ICCV)*, 2009.

[3] C. Rother, V. Kolmogorov, and A. Blake, “Grabcut: Interactive foreground extraction using iterated graph cuts,” *ACM Trans. Graph. (SIGGRAPH)*, 2004.

[4] L. Grady, “Random walks for image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2006.

[5] J. Zhang, J. Zheng, and J. Cai, “A diffusion approach to seeded image segmentation,” In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2010.

[6] T. H. Kim, K. M. Lee, and S. U. Lee, “Generative Image Segmentation using Random Walks with Restart,” In *Proc. European Conference on Computer Vision (ECCV)*, 2008.

[7] V. Gulshan, C. Rother. A. Criminisi, A. Blake, and A. Zisserman, “Geodesic star convexity for interactive image segmentation,” In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2010.

[8] S. D. Jain and K. Grauman, “Predicting Sufficient Annotation Strength for Interactive Foreground Segmentation,” In *Proc. International Conference on Computer Vision (ICCV)*, 2013.

[9] D. Freedman and T. Zhang, “Interactive graph cut based segmentation with shape priors,” In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2005.

[10] A. Protiere and G. Sapiro, “Interactive image segmentation via adaptive weighted distances,” *IEEE Transactions on Image Processing (TIP)*, 2007.

[11] E. N. Mortensen and W. A. Barrett, “Interactive segmentation with intelligent scissors,” *Graphical Models and Image Processing*, 1998.

[12] L. Vincent and P. Soille, “Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 1991.

[13] M. Everingham, L. Van Gool, C. K. I.Williams, J.Winn, and A. Zisserman. “The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results.”

[14] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott. “A perceptually motivated online benchmark for image matting,” In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2009.