SALIENCY CUTS BASED ON ADAPTIVE TRIPLE THRESHOLDING

Shuzhen Li, Ran Ju, Tongwei Ren and Gangshan Wu

State Key Laboratory for Novel Software Technology
Collaborative Innovation Center of Novel Software Technology and Industrialization
Nanjing University, Nanjing 210023, China
{szli, juran}@smail.nju.edu.cn, {rentw, gswu}@nju.edu.cn

ABSTRACT

Salient object detection attracts much attention for its effectiveness in numerous applications. However, how to effectively produce a high quality binary mask from a saliency map, named *saliency cuts*, is still an open problem. In this paper, we propose a novel saliency cuts approach using unsupervised seeds generation and GrabCut algorithm. With the input of a saliency map, we produce seeds for segmentation using adaptive triple thresholding, and feed the seeds to GrabCut algorithm. Finally, a high quality object mask is generated by iteratively optimization. The experimental results show that the proposed approach is competent to the task of saliency cuts and outperforms the state-of-the-art methods.

Index Terms— saliency cuts, saliency map, adaptive triple thresholding, GrabCut

1. INTRODUCTION

Salient object detection aims to detect the visual salient objects from an image [1]. It is widely used in numerous applications, including image classification [2], object retrieve [3], scene understanding [4], and image editing [5].

Generally speaking, salient object detection includes two steps, saliency map generation and saliency cuts [1]. Saliency map generation focuses on providing a pixel-level or region-level map (Fig. 1(b)) with the same size to the input image (Fig. 1(a)), in which each pixel or region has a value to denote its saliency, and the pixels with higher values have higher saliency. And saliency cuts focuses on providing a binary mask of salient object(s) (Fig. 1(c)) with the assistance of saliency map. There have been amounts of saliency map generation methods proposed, such as [6, 7], but saliency cuts is still an open problem. One solution for saliency cuts is calculating a threshold and decomposing the pixels in saliency map into two parts according to their saliency values [8, 9]. The effectiveness of this solution usually suffers from the inaccuracy of saliency map. To reduce the influence of the inaccuracy of saliency map, some methods utilize the input image and saliency map together in segmentation. [1, 10].

(a) (b) (c)

Fig. 1. (a) Input image. (b) Saliency map. (c) Saliency cuts.

Saliency cuts methods using input image and saliency map can be treated as a special object segmentation task, which is driven by saliency map. Object segmentation aims to decompose a input image into object and background using the features of color, texture, shape and so on [11]. The existing object segmentation methods can be roughly classified into three categories, learning based methods [12, 13], interactive methods [14, 15, 16], and automatic methods [17, 18]. Learning based methods require accurately labeled masks for training. They usually take enormous manual labor, and their performances strongly depend on prior knowledge. Interactive methods require skillful human interactions to provide foreground and background seeds. They require human interaction in the procedure of segmentation, and the segmentation results rely on the quality of seeds. Automatic methods do not require previous learning or human interaction, which are preferred in large scale and automatic applications. Yet their effectiveness is inferior to learning based methods and interactive method when the object(s) and background are complex.

Considering that saliency object detection is widely utilized in numerous applications, the procedure of saliency cuts should be automatical without previous learning or human interaction. However, different to the traditional automatic object segmentation methods, saliency cuts takes saliency map as an input, which provides effective supervision to segmentation. By analyzing saliency map, it is practicable to provide the required inputs of learning based methods or interactive based methods, and leverage their

This work is supported by the National Science Foundation of China (No.61321491, 61202320), Research Project of Excellent State Key Laboratory (No.61223003), Natural Science Foundation of Jiangsu Province(No.BK2012304), and National Special Fund (No.2011ZX05035-004-004HZ).

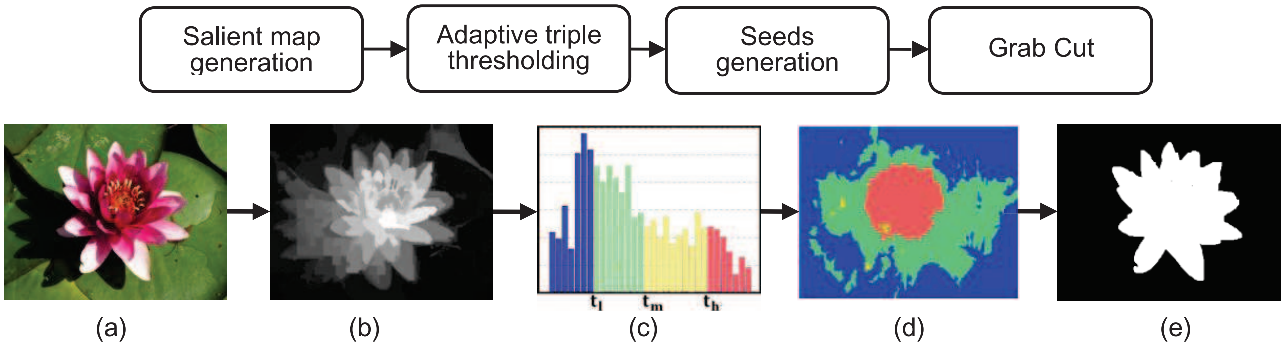

Fig. 2. The framework of our approach. (a) Input image. (b) Saliency map. (c) Adaptive triple thresholding on saliency histogram. (d) Seeds for segmentation. (e) Binary mask of salient object.

fruitful achievements in the automatic procedure of saliency cuts. The key problem is how to provide professional inputs by analyzing saliency maps and generate high quality binary masks effectively.

In this paper, we propose a novel approach to obtain professional seeds from a saliency map and segment the salient objects automatically. Firstly, a saliency map is generated using an existing saliency analysis method [1]. Then, using adaptive triple thresholding, the saliency map is marked as a four-regions seeds image. The seeds are fed to GrabCut algorithm [19] and a high quality binary mask is finally generated.

The proposed approach has the following contributions:

- We propose a novel seed generation method using adaptive triple thresholding saliency cuts.
- We integrate GrabCut algorithm with the auto-generated seeds to improve the segmentation result.

The remainder of this paper is organized as follows. In Section 2, we present the proposed saliency cuts based on adaptive triple thresholding. In Section 3, the experimental results are presented and discussed. Finally, the conclusion is given in Section 4.

2. ADAPTIVE TRIPLE THRESHOLDING BASED SALIENCY CUTS

Fig. 2 shows the framework of the proposed saliency cuts approach. First, a saliency map (Fig. 2(b)) is generated from the input color image (Fig. 2(a)) with an existing salient object detection method. Then, adaptive triple thresholding is implemented based on the histogram of saliency map, and three thresholds t_l, t_m, t_h are calculated by optimization to divide the saliency histogram into four parts (Fig. 2(c)). With the three thresholds, the saliency map is labelled with four different colors, which represents the seeds of different kinds (Fig. 2(d)). Finally, the seeds are fed to GrabCut algorithm, and a binary mask of salient object is generated by segmentation (Fig. 2(e)).

2.1. Adaptive Triple Thresholding

Otsu algorithm is an outstanding automatic thresholding algorithm in image segmentation for its simplicity and good performance. Based on clustering, Otsu algorithm automatically obtains an optimal threshold from the gray histogram of an original gray image and divide it into foreground and background [8]. The threshold maximizes the inter-class gray level difference between foreground and background, and minimizes the intra-class gray level difference within each part. And Otsu algorithm shows that minimizing the intra-class difference is the same problem of maximizing the inter-class difference.

Assume that the scale of a saliency map is $[0, H]$, and the pixel number of each bin in its histogram is $N = \{n_0, ..., n_i, ...n_H\}$, here i represents the saliency value. We improve Otsu algorithm to obtain three level thresholds, including t_l, t_m and t_h ($0 < t_l < t_m < t_h < H$), and decompose the histogram of a saliency map into the following four parts: certain foreground background $T_{cb} = [0, t_l]$, probable background $T_{pb} = [t_l + 1, t_m]$, probable foreground $T_{pf} = [t_m + 1, t_h]$, and certain foreground $T_{cf} = [t_h + 1, H]$. Different to Otsu multilevel thresholding method [20], we utilize twice optimizations to avoid the following QPSO algorithm. We first calculate t_m and divide the saliency histogram into $T_b = [0, t_m]$ and $T_t = [t_m + 1, H]$. Then, we further calculate t_l and t_h, and further divide the saliency histogram into T_{cb}, T_{pb}, T_{pf} and T_{cf}.

Assume that the numbers of pixels with saliency values in T_b and T_t are n_b and n_t, and the pixel number of the whole image is $n = n_b + n_t$. We calculate the threshold t_m by maximizing the inter-class difference between T_b and T_t:

$$t_m = \arg\max \sum \omega_t \omega_b (\mu_t - \mu_b)^2, \tag{1}$$

where ω_b and ω_t are the weights of T_b and T_t, which equals n_b/n and n_t/n, respectively; μ_b and μ_t are the mean saliency values of T_b and T_t, respectively, which are defined as:

$$\mu_k = \sum_{i \in T_k} \frac{in_i}{n_k}, k \in \{b, t\}. \tag{2}$$

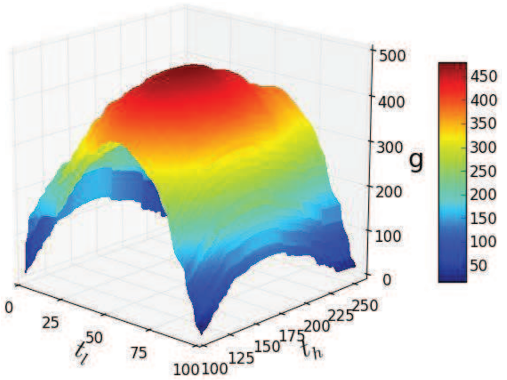

Fig. 3. The set of t_l, t_h and corresponding inter-class distance g with a given t_m.

t_l and t_h are further calculated as follows:

$$\{t_l, t_h\} = \arg\max \sum (\omega_{cb}\omega_{pb}(\mu_{cb} - \mu_{pb})^2 + \omega_{pf}\omega_{cf}(\mu_{pf} - \mu_{cf})^2), \quad (3)$$

where the weight $\omega_k = n_k/n_b$ when $k \in \{cb, pb\}$ and $\omega_k = n_k/n_t$ when $k \in \{pf, cf\}$; μ_k is the mean saliency value of each interval, which is defined as:

$$\mu_k = \sum_{i \in T_k} \frac{in_i}{n_k}, k \in \{cb, pb, pf, cf\}. \quad (4)$$

Given a saliency map and firstly find t_m that maximizes the difference between object region and background region. Then we step through all possible thresholds t_l and t_h and update ω_k and μ_k. The corresponding inter-class difference g is computed subsequently according to Eq.(3). Fig. 3 shows the procedure of finding t_l and t_h with given t_m. There is a set of $\{t_l, t_m, t_h\}$ maximizing the difference between four classes and that is what we need for seed generation.

According to the triple thresholded saliency histogram, saliency map can be also decomposed into four parts: certain background (pixels belong to T_{cb}), probable background (pixels belong to T_{pb}), probable foreground (pixels belong to T_{pf}) and certain foreground region (pixels belong to T_{cf}). Fig. 2(d) shows the seeds obtained by adaptive triple thresholding method in which the blue and green regions correspond to certain background and probable background respectively, the yellow region represents the probable foreground and the red region represents the certain foreground.

2.2. GrabCut with Auto-generated SeedS

Developing from interactive GraphCut algorithm, GrabCut algorithm represents an image by an graph with two terminals if given background regions as prior. In GrabCut algorithm the segmentation problem is formulated to find a mini-cut. The energy function is:

$$E(L, k, \theta, Z) = \sum_n D(l_n, k_n, \theta, z_n) + V(L, Z), \quad (5)$$

where Z is the gray value set of each pixel and L is the corresponding label set, θ is the gray histogram of foreground or background, and k is the parameter of GMM.

After introducing two GMM models, the data term D is defined as:

$$D(l_n, k_n, \theta, Z_n) = -\log \pi(l_n, k_n) + \frac{1}{2}\log\det \sum(l_n, k_n) + \frac{1}{2}[Z_n - \mu(l_n, k_m)]^T \sum(l_n, k_n)^{-1}[Z_n - \mu(l_n, k_n)]. \quad (6)$$

The smoothness term can be written as follows:

$$V(L, Z) = \gamma \sum_{(i,j)\in \mathbf{C}} [l_n \neq l_m] exp - \beta ||Z_m - Z_n||^2, \quad (7)$$

where $\beta = (2\langle(Z_m - Z_n)^2\rangle)^{-1}$ and $\langle\cdot\rangle$ in β denotes expectation over an colorful image.

With adaptive triple thresholding, we have already obtained the seeds of four kinds. Taking the certain background and probable background as background region T_b and initializing the label set L: $l_n = 0$ for $n \in T_b$ and $l_n = 1$ for $n \notin T_b$. From these labeled pixels, we can first calculate the parameters of two GMM modes, and then we get those pixels not belong to T_b relabeled by implementing iterative minimization algorithm. As a result, a new set L which can be proved to minimize the total energy E is obtained, and the image with two terminal is constructed.

3. EXPERIMENTS

3.1. Dataset and Experimental settings

We evaluate the proposed method on MSRA 10k dataset [1], which contains 10,000 images and corresponding manually labelled masks.

We implement our approach in C++ and carry out all the experiments on a machine with a 3.10GHz four-core CPU and 4GB memory. Typically for a 1280×720 image, the average running time for seeds generation and segmentation is 2.402s, excluding the time for saliency map generation.

3.2. Results Evaluation

We utilize region contrast algorithm [1] to generate saliency maps in our experiments. To demonstrate our results, we present some examples in Fig. 4. These images contain typical kinds of objects with various object numbers. As shown in the third row of Fig. 4, triple thresholding divides the input image into four parts with most salient pixels are set as foreground and least salient pixels are set as background.

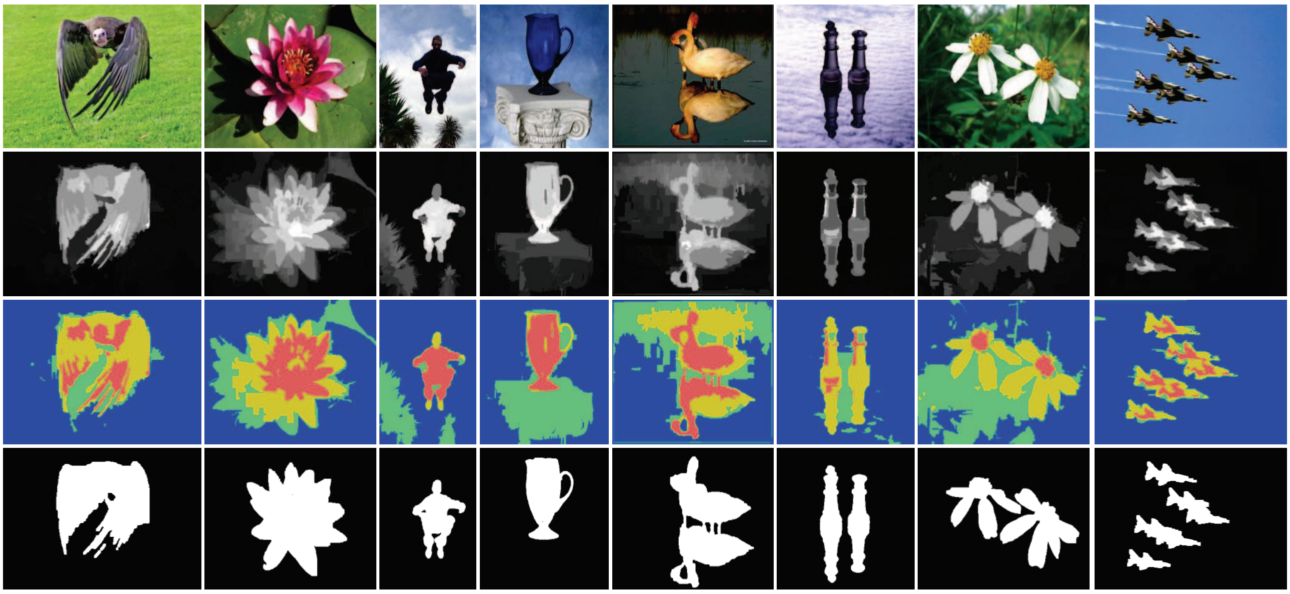

Fig. 4. Examples of saliency cuts results. (Top row) Input images. (Second row) Saliency maps. (Third row) Seeds generated by adaptive triple thresholding. (Bottom row) Our results.

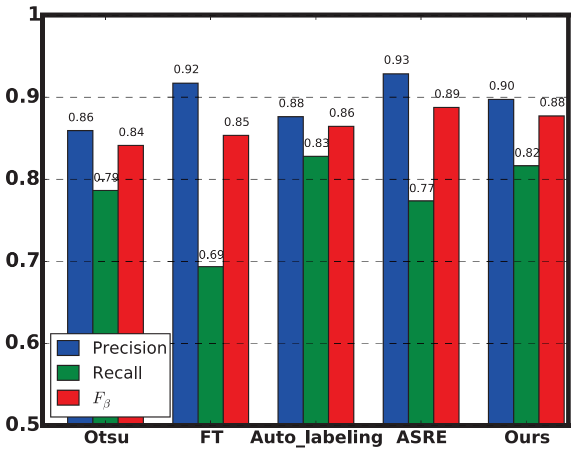

Fig. 5. Comparison with the state-of-the-art saliency cuts methods, including Otsu [8], FT [9], auto-labeling [10], and ASRE [1].

Fig. 6. Comparison of ASRE and our results. (Top row) Input images. (Second row) Saliency maps. (Third row) Results of ASRE method. (Bottom row) Our results.

For the seeds generated by our method provide more features of input images, our results are accurate and consistent.

We also validate our effectiveness by comparing our method with four state-of-the-art methods, including Otsu [8], frequency-tuned (FT) [9], auto-labeling [10], and automatic salient region extraction (ASRE) [1]. Fig.5 shows the precision, recall and F_β values of five methods on MSRA 10K datast, here $\beta^2 = 0.3$ [9]. It shows that the F_β value of our method is higher than other three methods except ASRE method. However, as show in Fig. 6, ASRE method may fail to detect salient objects when the saliency values of input image are low, and it can only detect at most one salient object in an image, which makes it miss some salient objects or parts of salient object in the segmentation results. Comparatively, our method can handle these situations and provide high quality binary masks.

4. CONCLUSION

We proposed a saliency cuts approach using unsupervised seeds generation and GrabCut algorithm. The approach automatically provides four kinds of seeds based on saliency map generation and adaptive triple thresholding, and feeds the seeds to GrabCut algorithm to obtain the binary masks of salient objects. The experiments demonstrate that our approach can obtain high quality results of automatic salient object segmentation and outperform the existing methods.

5. REFERENCES

[1] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu, "Global contrast based salient region detection," *TPAMI*, vol. 37, no. 3, pp. 409–416, 2015.

[2] Soo Beom Park, Jae Won Lee, and Sang Kyoon Kim, "Content-based image classification using a neural network," *PR*, vol. 25, no. 3, pp. 287–300, 2004.

[3] Xiangyang Xu, Wenjing Geng, Ran Ju, Yang Yang, Tongwei Ren, and Gangshan Wu, "Obsir: Object-based stereo image retrieval," in *ICME*. IEEE, 2014, pp. 1–6.

[4] Li-Jia Li, Richard Socher, and Li Fei-Fei, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework," in *CVPR*. IEEE, 2009, pp. 2036–2043.

[5] Brian Price and William Barrett, "Object-based vectorization for interactive image editing," *TVC*, vol. 22, no. 9-11, pp. 661–670, 2006.

[6] Laurent Itti, Christof Koch, and Ernst Niebur, "A model of saliency-based visual attention for rapid scene analysis," *TPAMI*, vol. 20, no. 11, pp. 1254–1259, 1998.

[7] Ali Borji, Dicky N Sihite, and Laurent Itti, "Salient object detection: A benchmark," in *ECCV*, pp. 414–429. Springer, 2012.

[8] N Otsu, "A threshold selection method from gray-level histograms," *TSMC*, vol. 9, no. 1, pp. 62–66, 1979.

[9] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk, "Frequency-tuned salient region detection," in *CVPR*. IEEE, 2009, pp. 1597–1604.

[10] Yu Fu, Jian Cheng, Zhenglong Li, and Hanqing Lu, "Saliency cuts: An automatic approach to object segmentation," in *ICPR*. IEEE, 2008, pp. 1–4.

[11] Liyuan Li, Weimin Huang, IY-H Gu, and Qi Tian, "Statistical modeling of complex backgrounds for foreground object detection," *TIP*, vol. 13, no. 11, pp. 1459–1472, 2004.

[12] Toshio Uchiyama and Michael A. Arbib, "Color image segmentation using competitive learning," *TPAMI*, vol. 16, no. 12, pp. 1197–1206, 1994.

[13] Bastian Leibe, Ales Leonardis, and Bernt Schiele, "Combined object categorization and segmentation with an implicit shape model," in *ECCV*. 2004, vol. 2, p. 7, Springer.

[14] Yuri Y Boykov and M-P Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *ICCV*. IEEE, 2001, pp. 105–112.

[15] Leo Grady, "Random walks for image segmentation," *TPAMI*, vol. 28, no. 11, pp. 1768–1783, 2006.

[16] Gergana Angelova Lazarova, "Semi-supervised image segmentation," in *AIMSA*, pp. 59–68. Springer, 2014.

[17] Jianbo Shi and Jitendra Malik, "Normalized cuts and image segmentation," *TPAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[18] John Winn and Nebojsa Jojic, "Locus: Learning object classes with unsupervised segmentation," in *ICCV*. IEEE, 2005, pp. 756–763.

[19] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *TOG*, vol. 23, no. 3, pp. 309–314, 2004.

[20] Yourui Huang and Shuang Wang, "Multilevel thresholding methods for image segmentation with otsu based on qpso," in *CISP*. IEEE, 2008, pp. 701–705.