Pattern Anal Applic (2015) 18:667–675
DOI 10.1007/s10044-015-0459-1

INDUSTRIAL AND COMMERCIAL APPLICATION

Extraction of salient objects based on image clustering and saliency

In Seop Na · Ha Le · Soo Hyung Kim · Guee Sang Lee · Hyung Jeong Yang

Received: 7 August 2013 / Accepted: 6 February 2015 / Published online: 20 February 2015

Abstract Over the past decades, numerous methods have been proposed on salient object detection. However, most of these methods need users' interactions as a prerequisite to control their progress. In this paper, we propose a novel method for extraction of salient objects based on image clustering and saliency map from natural scene images. This method is a combination of image clustering, saliency map generation and automatic initialization. First, a graph based clustering method is applied to split the input image into regions. Second, a saliency map of the input image is generated using the contrast among split regions. From the split regions and generated saliency map, an adaptive threshold is defined, which classify the split regions into foreground and background. After that, the initial mask for object detection is determined using the classified foreground and background clusters and saliency values. A grab-cut with our initial mask is applied to extract the objects of interest, and the experimental results have shown that our proposed method is able to replace manual labeling of initialization in object detection.

Keywords Automatic initialization · Grab-cut · Graph based clustering · Saliency map

1 Introduction

Salient object detection is one of the most fundamental functions in image processing and computer vision. It is the foundational tool in various advanced systems, such as object recognition [1], image retrieval [2], image editing [3] and scene reconstruction [4], etc. Because the results strongly influence the final results of these advanced systems salient detection, object segmentation has been an active research interest of many researchers on image processing and computer vision. Over the past decades, numerous methods have been proposed on salient object detection. However, most of these methods need some users' interactions as a prerequisite to control their progress.

Mortensen et al. [5] proposes an interactive method using global graph search, which allows users to choose minimum cost contour in an image. Bayes matting [6] requires a user-specified trimap, which separates an image into foreground, background, and unknown regions. The watershed transform [7], which finds "catchment basins" and "watershed ridge lines" in an image by treating it as a surface where light pixels are high and dark pixels are low, need the initial markers to control the over-segmentation. In the graph cut algorithms [8, 9], hard constraints are imposed by the user to provide seed positions and the goal is to find a minimum cost cut among all results satisfying the given hard constraints. Rother et al. [10] extends the graph cut approaches to simplify user interaction. In this method, the user specifies a rectangle loosely around an object. The active contour models [11] find object boundaries by iterative optimization. They make it possible to solve complicated and ill-posed object detection problems by combining priori boundary shape information with the image itself, and an initial shape given by the user. Although the user interaction-based methods are promising, they all pose a critical problem—the requirement of users' semantic intention. Moreover, the object detection performance heavily depends on user-specified seed locations. Thus, additional interactions are necessary when the seeds are not accurately identified.

I. S. Na · H. Le · S. H. Kim (✉) · G. S. Lee · H. J. Yang
School of Electronics and Computer Engineering, Chonnam National University, Gwangju 500-757, Korea
e-mail: shkim@jnu.ac.kr

To address this issue, several methods have been proposed to automatically initialize the objects of interest based on saliency concepts. Achanta et al. [12] extracts the objects of interest by applying an adaptive threshold to the saliency map of the input image. Since the ideal of Achanta's method is simple, its processing speed is fast. However, the accuracy is not sufficiently high for extraction purpose. Recently, Cheng [13] has proposed a method based on saliency map and grab-cut. The accuracy of Cheng's method is remarkable. However, this method employs grab-cut iteratively. Thus, its processing speed is slow and unstable, depending on the number of iterations.

To strike a balance between detection accuracy and processing time, we propose a novel method based on image clustering, saliency map generation and automatic initialization. Since visual saliency is the perceptual quality that highlights an object, person, or pixel from its neighbors and thus capturing our attention, we attempt to detect visually salient regions. To skip the manual initialization step from the algorithm, we determine the initializations for the salient objects based on salient regions and clusters obtained by an image clustering method. Since the initializations for the detection algorithms play a key role in the final results, selecting salient regions and determining appropriate initializations are the key points of our proposed method.

The rest of this paper is organized into four sections as follows. Section 2 presents related works. Section 3 presents detailed implementation of our proposed method with image clustering, saliency map generation and initialization. Section 4 discusses the performance and addresses some advantages of our method in comparison with current popular methods. Finally, a conclusion is given in Sect. 5.

2 Related works

The term saliency was used by Tsotsos et al. [15] and Olshausen et al. [16] in their work on visual attention, and by Itti et al. [17] in his work on rapid scene analysis. Saliency has also been referred to as visual attention [15, 18], unpredictability, rarity or surprise [19, 20]. Saliency estimation methods can broadly be classified as biologically based, purely computational or a combination of both aspects. In general, all methods employ a low-level approach by determining contrast of image regions relative to their surroundings, using one or more features of intensity, color and orientation [24–29].

The first category of methods is heavily influenced by biological principles. Based on the early representation model introduced by Koch and Ullman [30], Itti et al. [17] defines image saliency using center-surrounded differences across multi-scale image features. Frintrop et al. [31] presents a method inspired by Itti's method, but they compute center-surround differences with square filters and use integral images to speed up the calculations.

The second category of methods is purely computational and independent of any biological principles. Ma and Zhang [18] and Achanta et al. [12] estimate saliency using center-surround feature distances. Gao and Vasconcelos [33] maximize the mutual information between the feature distribution of central and surrounding regions in an image. Hu et al. [32] estimates saliency by applying heuristic measures on initial saliency measures obtained by histogram thresholding of feature maps, while Hou and Zhang [34] rely on the processing of frequency domain.

Harel et al. [35] detects saliency based on both aspects, biological and purely computational. He creates feature map using Itti's method but performs their normalization using a graph based approach. Bruce et al. [36] used a computational approach like the maximization of information that represents a biologically plausible model of saliency detection.

One of the limitations of most saliency detection methods is the resolution of saliency map. Itti's method [17] produces a saliency map that is just 1/256th the original image size, while Hou and Zhang [34] output maps of size 64×64 pixels for any input image size. Because of the downsized input image, some salient region detectors have ill-defined object boundaries [17, 31, 35]. Besides, other methods highlight the salient object boundaries, but fail to uniformly map the entire salient region [18, 34] or highlight smaller salient regions better than larger ones [12]. These problems are overcome in two methods proposed by Cheng et al. [13], namely histogram based contrast and region based contrast.

3 Proposed method

Our proposed method consists of four main steps: saliency map generation, image clustering, initialization and salient object detection. Figure 1 shows the flowchart of the proposed method.

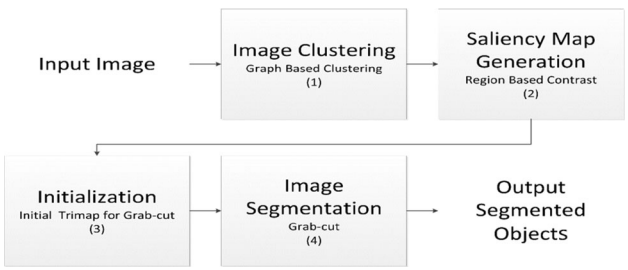


Fig. 1 Flowchart of the proposed method

3.1 Image clustering

In the proposed method, an image clustering method is applied to obtain image clusters, which will be used to generate saliency map and determine the initial mask for grab-cut segmentation method in the next steps. There are several image clustering methods in the literature, such as *k*-means, Gaussian mixture model (GMM), mean shift or graph based clustering. However, *k*-means and GMM need a specific number of clusters as an input parameter, while mean shift requires much processing time. Thus, in this paper, graph based clustering method is chosen due to its flexible input parameters and its fast processing speed.

Graph based image clustering techniques generally represent the problem in terms of a graph $G = (V, E)$ where each node $v_i \in V$ corresponds to a pixel in the image, and each edge $(v_i, v_j) \in E$ corresponds to pairs of neighboring vertices. A weight $w(v_i, v_j)$ is associated with each edge based on some property of the pixels that it connects, such as their image intensities.

In the graph based approach, a segmentation S is a partition of V into components such that each component (or region) $C \in S$ corresponds to a connected component in a graph $G' = (V, E')$, where $E' \subseteq E$. In other words, any segmentation is induced by a subset of the edges in E. There are different ways to measure the quality of segmentation but in general we want the elements in a component to be similar, and elements in different components to be dissimilar. This means that edges between two vertices in the same component should have relatively low weights, and edges between vertices in different components should have higher weights.

An efficient graph based clustering algorithm is introduced by Felzenszwalb and Huttenlocher et al. [22]. The detailed process of their algorithm is as follows:

The input is a graph $G = (V, E)$, with n vertices and m edges. The output is a segmentation of V into components $S = (C_1, C_2,\ldots,C_r)$.

Step 0: Sort E into $\pi = (o_1, o_2,\ldots,o_m)$, by non-decreasing edge weight.
Step 1: Start with a segmentation S^0, where each vertex v_i is in its own component.
Step 2: Repeat step 3 for $q = 1,\ldots,m$.
Step 3: Construct S^q given S^{q-1}as follows. Let v_i and v_j denote the vertices connected by the qth edge in the ordering, i.e., $o_q = (v_i, v_j)$. If v_i and v_j are in disjoint components of S^{q-1} and $w(o_q)$ is small compared to the internal difference of both those components, merge the two components. Otherwise, the two components remain intact. More formally, let C_i^{q-1} be the component of S^{q-1}containing v_i and C_j^{q-1} the component containing v_j. If $C_i^{q-1} \neq C_j^{q-1}$ and $w(o_q) \leq \text{MInt}(C_i^{q-1}, C_j^{q-1})$ then S^q is obtained from S^{q-1} by merging C_i^{q-1} and C_j^{q-1}. Otherwise $S^q = S^{q-1}$.
Step 4: Return $S = S^m$.

In this algorithm, the internal difference of a component $C \subseteq V$ is the largest weight in the minimum spanning tree of the component, MST(C, E). That is,

$$\text{Int}(C) = \max_{e \in \text{MST}(C, E)} w(e). \quad (1)$$

The minimum internal different, MInt, is defined using the internal difference of two components,

$$\text{MInt}(C_1, C_2) = \min\begin{pmatrix} \text{Int}(C_1) + \tau(C_1), \\ \text{Int}(C_2) + \tau(C_2) \end{pmatrix}. \quad (2)$$

For small components, Int(C) is not a good estimate of the local characteristics of the data. Therefore, a threshold function based on the size of component is used,

$$\tau(C) = k/|C| \quad (3)$$

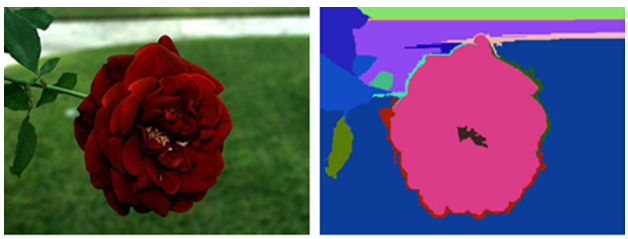
where $|C|$ denotes the size of C, and k is a constant parameter. In practice k sets a scale of observation, in which a larger k causes a preference for larger components.

Figure 2 shows an example of graph based clustering. In this example the constant k is set to 300, and the number of segmented clusters is 16.

3.2 Saliency map generation

Saliency estimation methods can broadly be classified as biologically based, purely computational or a combination of both aspects. In general, all methods employ a low-level approach by determining contrast of image regions relative to their surroundings, using one or more features of intensity, color and orientation.

Humans pay more attention to those image regions that contrast strongly with their surroundings [21]. Besides contrast, spatial relationships play an important role in human attention. The saliency of a region is more evident



(a) **(b)**

Fig. 2 An example of graph based clustering, **a** the input image and **b** the resulting clusters

with a high contrast to its surrounding than a high contrast to distant regions. Moreover, through our observation, the location of object of interest is mostly near the image center. Thus, in this paper, we combine the constraint that the location of object of interest is near the image center with the region based contrast (RC) method proposed by Cheng (Cheng, 2011) for saliency map generation.

3.2.1 Region based contrast

RC is a contrast analysis method that integrates spatial relationships into region-level contrast computation. In RC, the graph based image clustering method presented in the above section is first used to segment the input image into regions. The saliency value of a region r_k is computed by measuring its color contrast to all other regions in the image.

$$S(r_k) = \sum_{r_k \neq r_i} w(r_i) D_r(r_k, r_i) \tag{4}$$

where $w(r_i)$ is the weight of regions r_i and $D_r(r_k, r_i)$ is the color distance metric between the two regions. Here the number of pixels in r_i is assigned as $w(r_i)$ to emphasize color contrast to bigger regions. The color distance between two regions r_1 and r_2 is defined as,

$$D_r(r_1, r_2) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f(c_{1,i}) f(c_{2,j}) D(c_{1,i}, c_{2,j}) \tag{5}$$

where $f(c_{k,i})$ is the probability of the ith color $c_{k,i}$ among all n_k colors in the kth region r_k, $k = 1, 2$, and $D(c_{1,i}, c_{2,j})$ is the Euclidean distance between two colors, $c_{1,i}$ and $c_{2,j}$. The probability of a color in the probability density function of the region is used as the weight for this color to emphasize the color differences between dominant colors.

The computation time of Eq. 5 depends on the number of distinct colors in the input image. Therefore, reducing the number of colors is necessary to reduce the computation time. There are many solutions to reduce the number of distinct colors in an image, such as using only one channel in color space or using only intensity value, etc. In RC method, the number of distinct pixel colors is reduced in two steps.

Step 1: Quantize each color channel in RGB color space of the input image to 12 different values. After quantization, the number of distinct pixel colors in the input image is less than or equal to $12^3 = 1728$.
Step 2: Choose more frequently occurring colors and ensure these colors cover the colors of more than 95 % of the image pixels. The colors of the 5 % remaining pixels are replaced by the closest colors in the histogram.

Besides color information, the spatial information is also incorporated to increase the effects of closer regions and decrease the effects of farther regions. For any region r_k, the spatially weighted region contrast based saliency is defined as,

$$S(r_k) = \sum_{r_k \neq r_i} \exp\left(-D_s(r_k, r_i)/\sigma_s^2\right) w(r_i) D_r(r_k, r_i) \tag{6}$$

where $D_s(r_k, r_i)$ is the spatial distance between regions r_k and r_i, and σ_s controls the strength of spatial weighting. Larger values of σ_s reduce the effect of spatial weighting so that the contrast to farther regions would contribute more to the saliency of the current region. The spatial distance between two regions is defined as the Euclidean distance between their centroids.

3.2.2 Extended region based contrast

Based on the assumption that the location of object of interest is near the image center, we extend Eq. 6 with the distance from each region to the image center. Thus, the saliency of a region r_k is defined as,

$$S(r_k) = \sum_{r_k \neq r_i} \exp\left(-(D_s(r_k, r_i) + D_t(r_k, t))/\sigma_s^2\right) w(r_i) D_r(r_k, r_i) \tag{7}$$

where t is the image center, and $D_t(r_k, t)$ is the Euclidean distance from r_k to t.

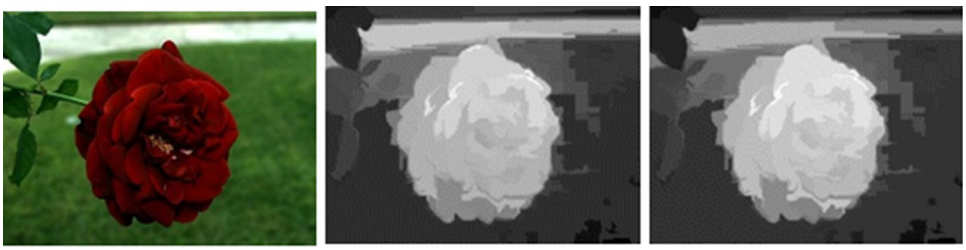
In our implementation, we first segment the input image into regions using graph based clustering method with $k = 50$. After that, we apply our extended region based contrast (ERC) method with $\sigma_s^2 = 0.4$ and the pixel coordinates normalized to [0, 1]. Figure 3 shows an example result of saliency map using the RC and ERC methods. Figure 3a shows the input image, Fig. 3b shows the result of RC method, and Fig. 3c shows the result of ERC method. The center pixels of Fig. 3c are lighter than those of Fig. 3b, and the pixels far from center of Fig. 3c are darker than those of Fig. 3b.

3.3 Initialization

In this paper, the grab-cut segmentation method is applied to segment the input image. Since grab-cut needs the initial trimap for segmentation, a method to determine the initial trimap for grab-cut is presented in this step. At first, the clusters obtained from the previous step are classified into background and foreground clusters using two adaptive thresholds. After that, these background and foreground clusters along with saliency map are used to determine the initial trimap.

3.3.1 Background and foreground classification

Suppose that the input image is split into K clusters after image clustering step. With each cluster $C_i (i = 1,\ldots,K)$, the average saliency value $V_i (i = 1,\ldots,K)$ is calculated by



(a) **(b)** **(c)**

Fig. 3 A saliency map example, **a** the input image and **b** the saliency map generated by RC method and **c** the saliency map generated by ERC method

adding up the values in the saliency map corresponding to pixels belong to the cluster,

$$V_i = \frac{1}{|C_i|}\sum_{I_k \in C_i} S(I_k) \tag{8}$$

where $|C_i|$ is the size of the cluster in pixels, I_k is an arbitrary pixel of the cluster C_i, and $S(I_k)$ is the saliency value of pixel I_k. An adaptive threshold based method is used to distinguish background clusters from foreground clusters. The clusters having average saliency value greater or equal to a threshold T_a are marked as foreground, while, the clusters having average saliency value less than the threshold T_a are marked as background. The adaptive threshold T_a is determined as,

$$T_a = \frac{\alpha}{|I|}\sum_{I_k \in I} S(I_k) \tag{9}$$

where $|I|$ is the size of the input image I in pixels, I_k an arbitrary pixel of I, $S(I_k)$ is the saliency value of pixel I_k, and α is a constant parameter that controls the number of clusters passing through the threshold. If α is large, there are fewer clusters passing through the threshold, and vice versa. When α is too large, there is no cluster passing through the threshold. Therefore, we choose an adaptive threshold that is lower than the highest average saliency value. Thus, at least one cluster is marked as foreground with the following T_a' threshold,

$$T_a' = \min\left(T_a, \max_{i=1,\ldots,K} V_i\right) \tag{10}$$

In contrast to a large value of α, a small value of α may increase the number of unexpected clusters being marked as foreground. To reduce these unexpected clusters, we rely on the constraint that most pixels of the object of interest are not placed on the boundary of the input image. First, we define a threshold T_b based on the number of pixels around the boundary of the input image,

$$T_b = \frac{|B|}{\gamma K} \tag{11}$$

where $|B|$ is the size of the boundary in pixels, and γ is a constant value controlling the magnitude of T_b. Suppose that the margin of the left and right boundaries is $W/16$, and the margin of the top and bottom boundaries is $H/16$ (Fig. 4e), where W and H are the width and height of the input image in pixels, the threshold T_b can be determined as,

$$T_b = \frac{15\ (W \times H)}{64\gamma K} \tag{12}$$

Assume that the number of pixels of cluster C_i placed on the boundary of the input image is B_i, cluster C_i is marked as foreground if,

$$V_i \geq T_a^{'} \text{ and } B_i \leq T_b \tag{13}$$

Otherwise, C_i is marked as background. To avoid the problem when T_b is too small, and there is no cluster passing through this threshold, we need to re-define T_b. With $C_f = \{C_i(i = 1,\ldots,K)|V_i \geq T_a'\}$ is the set of clusters passing through the threshold T_a', the threshold T_b can be re-defined as,

$$T_b^{'} = \max\left(T_b, \min_{B_i \in C_f} B_i\right) \tag{14}$$

Finally, a cluster C_i is marked as foreground if,

$$V_i \geq T_a^{'} \text{ and } B_i \leq T_b^{'} \tag{15}$$

Otherwise, C_i is marked as background.

With the assumption that the object of interest is solid, we remove holes inside the object of interest using connected component labeling algorithm.

Figure 4 shows an example of classifying foreground and background clusters. Figure 4a is the input image. Figure 4b is the clustering result using graph based clustering with $k = 300$, and Fig. 4c is the saliency map of the input image using ERC method. Figure 4d shows the remained clusters after applying threshold T_a' with $\alpha = 1.5$. Figure 4e shows that the violet cluster has many pixels placed on the boundary of the input image. Thus, when we apply T_b' threshold, this cluster is re-marked as background as shown in Fig. 4f. Finally, we obtained the Fig. 4g by filling all inner holes of the image (f).

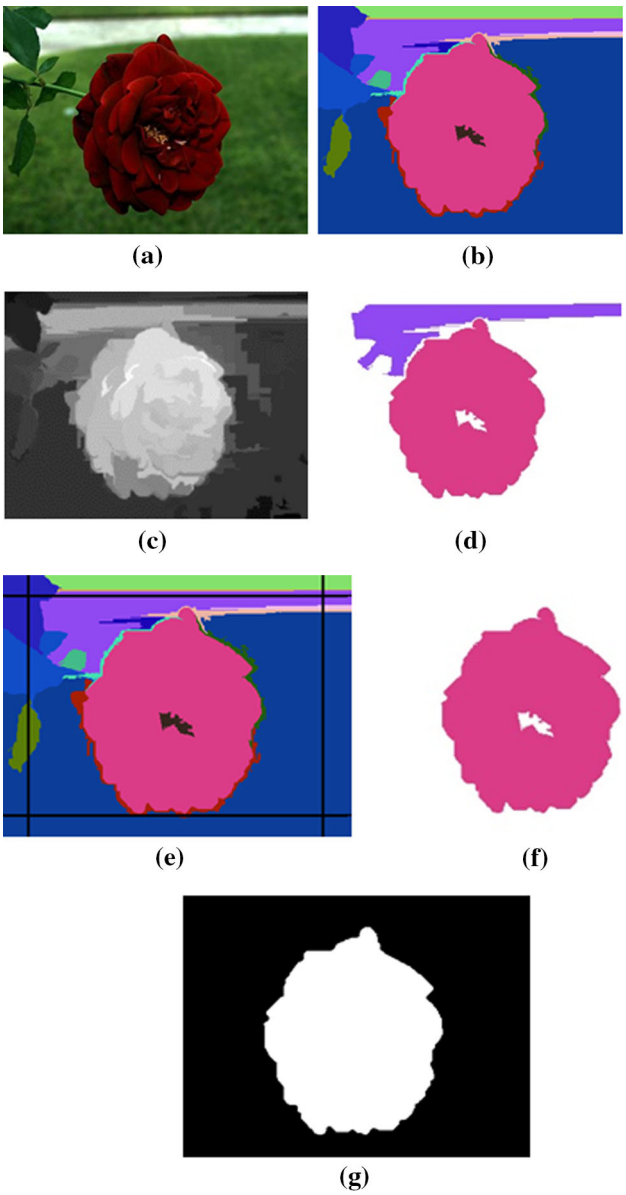


Fig. 4 Classifying foreground and background clusters, **a** the input image, **b** the segmented clusters, **c** the corresponding saliency map, **d** after applying threshold T_a' with $\alpha = 1.5$, **e** the image (**b**) with *black boundary lines*, **f** after classifying using both T_a' and T_b' thresholds with $\alpha = 1.5$ and $\gamma = 1.5$ and **g** the image (**f**) filled all inner holes

3.3.2 Initial trimap for grab-cut

To apply grab-cut algorithm, we need to create an initial trimap with four different types of values, which define the obvious background pixels, the obvious foreground pixels, the possible background pixels and the possible foreground pixels.

We notice that the saliency values of background pixels are normally lower than that of the object of interest. Thus, we mark the lowest saliency pixels in background cluster as the obvious background pixels in the initial trimap. The chosen pixels are ensured to cover 50 % of the background cluster, and the remaining pixels in the background cluster are marked as the possible background pixels in the initial trimap.

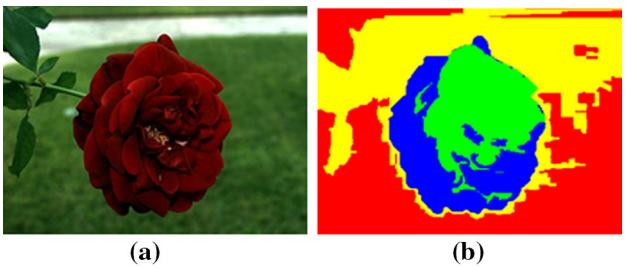


Fig. 5 The initial trimap for grab-cut, **a** the input image and **b** the initial mask

In contrast with background pixels, we mark the highest saliency pixels in the foreground clusters as the obvious foreground pixels in the initial trimap. The chosen pixels are ensured to cover 50 % of the foreground cluster, and the remaining pixels in the foreground cluster are marked as the possible foreground pixels in the initial trimap.

Figure 5 shows an example of initial trimap determination. Figure 5a shows the input image, and Fig. 5b shows the initial trimap. The red pixels are the obvious background pixels, the yellow pixels are the possible background pixels, the green pixels are the obvious foreground pixels and the blue pixels are the possible foreground pixels.

4 Experiments and results

4.1 Experimental results

The proposed method is tested on the publicly available database provided by Achanta et al. [12] to evaluate its performance. Achanta's database contains 1000 images and has ground truth in the form of manually annotated labels for object segmentation. However, only 800 images whose objects of interest are placed near their image centers are chosen as a test data set. To evaluate the performance of our proposed method, we compare our proposed method with manual initialization and three other automatic initialization methods, PV (Parvati's method) et al. [23], MGAC (morphological gradient applied to new active contour) et al. [14] and RCC (region based contrast and saliency cut) et al. [13] (Fig. 6).

The PV [23] method initializes the markers for watershed algorithm based on morphological operations. The MGAC [14] method determines the initial contour for level set algorithm as a rectangle in the image center. The RCC

Fig. 6 Grab-cut segmentation result

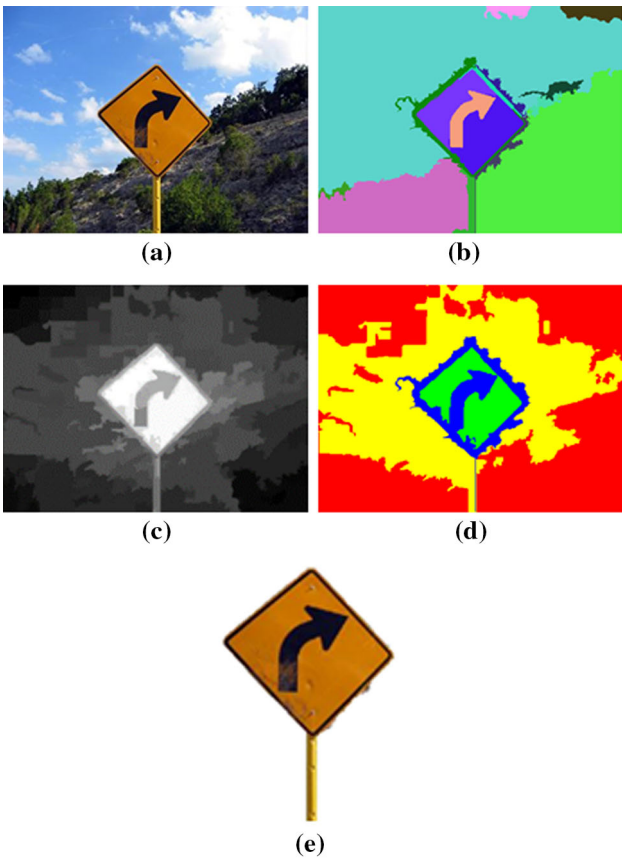


Fig. 7 An example result of the proposed method, **a** the input image, **b** the graph based clustering result, **c** the saliency map, **d** the initial trimap and **e** the extracted object

[13] method defines the initial rectangle for grab-cut algorithm using saliency map. To manual initial for grab-cut, we simply draw a rectangle around the image center. A visual comparison of the results is shown in Figs. 7, 8 and 9.

Figure 7 shows another example result of our proposed method. Figure 7a is the input image, whose object of interest is located in the image center. Figure 7b shows the graph based clustering result, and Fig. 7c shows the saliency map generated by ERC method. Figure 7d shows the initial trimap, and Fig. 7d shows the segmentation result.

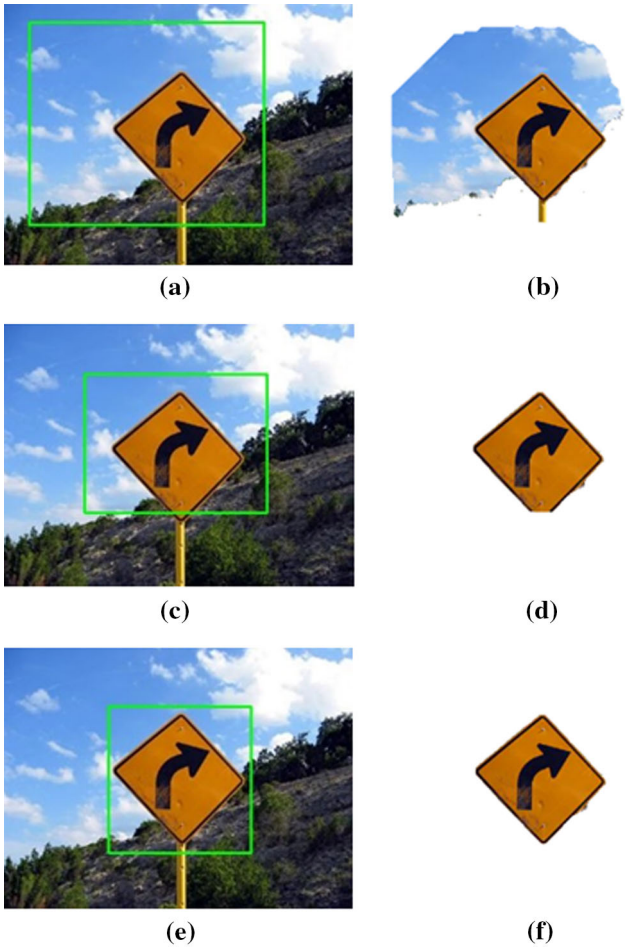


Fig. 8 Example results of manual initialization, **a**, **c** and **e** the input image with *three different initial rectangles* and **b**, **d** and **f** the grab-cut results with the *initial rectangles* in (**a**), (**c**) and (**e**), respectively

The examples in Fig. 8 show that the segmentation results of manual initialization vary according to how close the initialization to the optimal solution. If the user initializes near the object's boundary, the segmentation result is almost perfect. However, this task can be frustrating and boring. Since most users usually initialize with insufficient information, they sometimes miss certain parts of the object. In this case, the segmentation result may not as good as that of our automatic method.

Figure 9 shows the segmentation results of other automatic initialization methods. Figure 9a shows the segmentation result of PV [23] method, Fig. 9b shows the segmentation result of MGAC [14] method, and Fig. 9c shows the segmentation result of RCC [13] method.

(a) (b) (c)

Fig. 9 Example results of other automatic initialization methods, **a** PV [23] method, **b** MGAC [14] method and **c** RCC [13] method

Table 1 Accuracy for salient object detection

Methods	GCS	PV [23]	MGAC [14]	RCC [13]
Precision	91	55	65	90
Recall	89	67	72	90
F-measure	91	57	67	90

4.2 Interpretation

For a quantitative comparison, average precision (P), recall (R) and F-measure (F) are computed over the entire ground truth database, with precision, recall and F-measure defined as,

$$P = \frac{\text{Groundtruth} \cap \text{Segmented}}{\text{Segmented}} \tag{16}$$

$$R = \frac{\text{Groundtruth} \cap \text{Segmented}}{\text{Groundtruth}} \tag{17}$$

$$F_\beta = \frac{(1+\beta^2)\text{Precision} \times \text{Recall}}{\beta^2 \text{Precision} + \text{Recall}} \tag{18}$$

We use $\beta^2 = 0.3$ to weigh precision more than recall. A comparison of segmentation accuracies are shown in Table 1.

From the results shown in Table 1, our proposed method GCS (grab-cut with clustering and saliency map) outperforms the PV [23] and MGAC [14] methods. This proves that our proposed method is better than PV [23] and MGAC [14] methods. The segmentation result of our proposed method CSTS is comparable to the RCC [13] method. However, RCC [13] method uses a fixed threshold to binarize the saliency map. Thus, the segmentation result of RCC [13] method may vary depending on the threshold value. The segmentation result of RCC [13] method shown in Table 1 is the best one selected after testing on the proposed database with threshold value 70.

Regarding processing time, our GCS method has proved its advantage in comparison to RCC [13] method. The average processing times in millisecond of four methods are shown in Table 2. The testing environment is CPU Core i3 3.10 GHz and 3 GB RAM. All the programs are written in C++.

Table 2 Average processing times

Methods	GCS	PV [23]	MGAC [14]	RCC [13]
Time (ms)	828	182	>5000	1520

Since our proposed GCS method uses salient object detection methods with the initialization almost matching the final result, it is able to converge to an optimal solution earlier than the salient object detection methods with arbitrary initialization.

5 Conclusions

Object segmentation is carried out as a foundational step in advanced image processing techniques and computer vision. To exclude user interaction and segment the object of interest automatically, we proposed a novel method based on saliency map, image clustering and salient object detection. By using saliency map and split clusters from an image clustering method as the initial information, the initial trimap for grab-cut segmentation method is automatically determined. The segmentation results have shown that our proposed method can replace the tedious task of manual labeling for grab-cut segmentation method. Its computational time is also considerably saved. In the future, we will combine more prior information of the object of interest to improve the segmentation accuracy of the proposed method.

Acknowledgments This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2013-022495).

References

1. Fussenegger M, Opelt A, Pinz A, Auer P (2004) Object recognition using segmentation for feature detection. In: Proc. IEEE int. conf. pattern recognition, pp 41–44
2. Hirata K, Kasutani E, Hara Y (2002) On image segmentation for object-based image retrieval. In: Proc, IEEE int. conf. pattern recognition, pp 1031–1034
3. Barrett WA, Cheney AS (2002) Object-based image editing. ACM Trans Graph 21(3):777–784
4. Agrawal AK, Chellappa R (2005) Moving object segmentation and dynamic scene reconstruction using two frames. ICASSP 2:705–708
5. Mortensen EN, Barrett WA (1995) Intelligent scissors for image composition. In: Proc. of ACM SIGGRAPH, pp 191–198
6. Chuang YY, Curless B, Salesin DH, Szeliski R (2001) A bayesian approach to digital matting. In: Proc. of IEEE international conference on computer vision and pattern recognition, pp 264–271
7. Meyer F (1994) Topographic distance and watershed lines. Sig Process 38:113–125

8. Boykov Y, Jolly MP (2001) Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In: Proc. IEEE int. conf. computer vision, pp 105–112
9. Boykov Y, Funka-Lea G (2006) Graph cuts and efficient N-D image segmentation. Int J Comput Vis 70(2):109–131
10. Rother C, Kolmogorov V, Blake A (2004) Grabcut-interactive foreground extraction using iterated graph cuts. In: Proc. ACM SIGGRAPH, pp 309–314
11. Kass M, Witkin A, Terzopoulous D (1987) Snakes: active contour models. Int J Comput Vis 1:321–331
12. Achanta R, Hemami S, Estrada F, Susstrunk S (2009) Frequency-tuned salient region detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, pp 1597–1604
13. Cheng MM, Zhang GX, Mitra NJ, Huang X, Hu SM (2011) Global contrast based salient region detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, pp 409–416
14. Anh NTL, Kim YC, Lee GS (2012) Morphological gradient applied to new active contour model for color image segmentation. In: Proceedings of the 6th international conference on ubiquitous information management and communication Malaysia, (CD-pub)
15. Tsotsos JK, Culhane SM, Wai WYK, Lai Y, Davis N, Nuflo F (1995) Modeling visual attention via selective tuning. Artif Intell 78(1–2):507–545
16. Olshausen B, Anderson C, Van Essen D (1993) A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. J Neurosci 13:4700–4719
17. Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell 20(11):1254–1259
18. Ma YF, Zhang HJ (2003) Contrast-based image attention analysis by using fuzzy growing. ACM international conference on multimedia
19. Kadir T, Zisserman A, Brady M (2004) An affine invariant salient region detector. European conference on computer vision
20. Itti L, Baldi PF (2005) Bayesian surprise attracts human attention. Adv Neural Inf Process Syst 19:547–554
21. Eihhauser W, Konig P (2003) Does luminance-contrast contribute to a saliency map for overt visual attention? Eur J Neurosci 17:1089–1097
22. Felzenszwalb P, Huttenlocher D (2004) Efficient graph-based image segmentation. Int J Comput Vis 59(2):167–181
23. Parvati K, Prakasa Rao BS, Mariya Das M (2008) Image segmentation using gray-scale morphology and marker-controlled watershed transformation. Discret Dyn Nat Soc J 1–8
24. Chan AB, Vasconcelos N (2008) Modeling, clustering, and segmenting video with mixtures of dynamic textures. IEEE Trans Pattern Anal Mach Intell 30(5):909–926
25. Chen J, Zhao G, Salo M, Rahtu E, Pietikäinen M (2013) Automatic dynamic texture segmentation using local descriptors and optical flow. IEEE Trans Image Process 22(1):326–339
26. Xie Y, Lu H, Yang MH (2013) Bayesian saliency via low and mid level cues. IEEE Trans Image Process 22(5):1689–1698
27. Mishray A, Aloimonosy Y, Fah CL (2009) Active segmentation with fixation. In: Computer vision, 2009 IEEE 12th international conference, pp 468–475
28. Liu W, Tao D (2013) Multiview hessian regularization for image annotation. IEEE Trans Image Process 22(7):2676–2687
29. Liu W, Tao D, Cheng J, Tang Y (2014) Multiview hessian discriminative sparse coding for image annotation. Comput Vis Image Underst 118:50–60
30. Koch C, Ullman S (1985) Shifts in selective visual attention: towards the underlying neural circuitry. Human Neurobiol 4(4):219–227
31. Frintrop S, Klodt M, Rome E (2007) A real-time visual attention system using integral images. International conference on computer vision systems
32. Hu Y, Xie X, Ma WY, Chia LT, Rajan D (2004) Salient region detection using weighted feature maps based on the human visual attention model. Pacific Rim conference on multimedia
33. Gao D, Vasconcelos N (2007) Bottom-up saliency is a discriminant process. IEEE conference on computer vision
34. Hou X, Zhang L (2007) Saliency detection: a spectral residual approach. IEEE conference on computer vision and pattern recognition
35. Harel J, Koch C, Perona P (2007) Graph-based visual saliency. Adv Neural Inf Process Syst 19:545–552
36. Bruce N, Tsotsos J (2007) Attention based on information maximization. J Vis 7(9):950