Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Interactive image segmentation by maximal similarity based region merging☆

Jifeng Ning[a,b,c], Lei Zhang[a,*], David Zhang[a], Chengke Wu[b]

[a] Biometric Research Center, Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong, China
[b] State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, China
[c] College of Information Engineering, Northwest A & F University, Yangling, China

ARTICLE INFO

Article history:
Received 1 June 2008
Received in revised form 3 January 2009
Accepted 2 March 2009

Keywords:
Image segmentation
Region merging
Maximal similarity
Mean shift

ABSTRACT

Efficient and effective image segmentation is an important task in computer vision and object recognition. Since fully automatic image segmentation is usually very hard for natural images, interactive schemes with a few simple user inputs are good solutions. This paper presents a new region merging based interactive image segmentation method. The users only need to roughly indicate the location and region of the object and background by using strokes, which are called markers. A novel maximal-similarity based region merging mechanism is proposed to guide the merging process with the help of markers. A region R is merged with its adjacent region Q if Q has the highest similarity with Q among all Q's adjacent regions. The proposed method automatically merges the regions that are initially segmented by mean shift segmentation, and then effectively extracts the object contour by labeling all the non-marker regions as either background or object. The region merging process is adaptive to the image content and it does not need to set the similarity threshold in advance. Extensive experiments are performed and the results show that the proposed scheme can reliably extract the object contour from the complex background.

1. Introduction

Image segmentation is to separate the desired objects from the background. In general, the color and texture features in a natural image are very complex so that the fully automatic segmentation of the object from the background is very hard. Therefore, semi-automatic segmentation methods incorporating user interactions have been proposed [2,4,10,13,17,20,21,24] and are becoming more and more popular. For instance, in the active contour model (ACM), i.e. the snake algorithm [2], a proper selection of the initial curve by the user could lead to a good convergence to the true object contour. Similarly, in the graph cut algorithm [10–12], the prior information obtained by the users is critical to the segmentation performance.

The low level image segmentation methods, such as mean shift [5,6], watershed [3], level set [15] and super-pixel [28], usually divide the image into many small regions. Although may have severe over segmentation, these low level segmentation methods provide a good basis for the subsequent high level operations, such as region merging. For example, in [17,18], Li et al. combined graph cut with watershed pre-segmentation for better segmentation outputs, where the segmented regions by watershed, instead of the pixels in the original image, are regarded as the nodes of graph cut. As a popular segmentation scheme for color image, mean shift [6] can have less over segmentation than watershed while preserving well the edge information of the object (Fig. 1a shows an example). Because of less over segmentation, the statistic features of each region, which will be exploited by the proposed region merging method, can be more robustly calculated and then be used in guiding the region merging process.

In this paper, we proposed a novel interactive region merging method based on the initial segmentation of mean shift. In the proposed scheme, the interactive information is introduced as markers, which are input by the users to roughly indicate the position and main features of the object and background. The markers can be the simple strokes (e.g. the green and blue lines in Fig. 1b). Then the proposed method will calculate the similarity of different regions and merge them based on the proposed maximal similarity rule with the help of these markers. The object will then be extracted from the background when the merging process ends (Fig. 1c shows an example of segmentation result).

Although the idea of introducing markers into interactive segmentation was used in Meyer's watershed scheme [4] and the graph cut schemes [10–12], this paper first uses it to guide the region merging for object contour extraction. The key contribution of the proposed method is a novel maximal similarity based region merging (MSRM) mechanism, which is adaptive to image content and does

☆ This work is partially supported by the National Science Foundation Council of China under Grants 60532060 and 60775020.

* Corresponding author.
E-mail addresses: jf_ning@sina.com (J. Ning), cslzhang@comp.polyu.edu.hk (L. Zhang), csdzhang@comp.polyu.edu.hk (D. Zhang), ckwu@xidian.edu.cn (C. Wu).

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.03.004

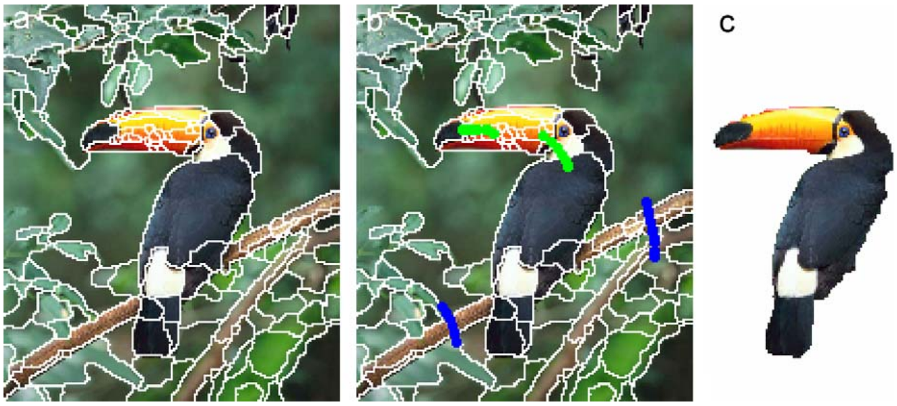

Fig. 1. (a) Initial mean shift segmentation. (b) The interactive information input by the user. The green line is the object marker and the blue lines are the background markers. (c) Segmentation result by the proposed region merging method. For interpretation of the references to color in this figure legend, the reader is referred to the webversion of this article.

not require a preset threshold. With the proposed region merging algorithm, the non-marker background regions will be automatically merged and labeled, while the non-marker object regions will be identified and avoided from being merged with background. Once all the non-marker regions are labeled, the object contour can then be readily extracted from the background. The proposed algorithm is very simple but it can successfully extract the objects from complex scenes.

The rest of the paper is organized as follows. Section 2 presents the proposed region merging algorithm. Section 3 performs extensive experiments to verify the proposed method. Section 4 concludes the paper.

2. Maximal-similarity based region merging

In our method, an initial segmentation is required to partition the image into homogeneous regions for merging. Any existing low level segmentation methods, such as super-pixel [28], mean shift [5,6], watershed [3] and level set [15], can be used for this step. In this paper, we choose to use the mean shift method for initial segmentation because it has less over segmentation and can well preserve the object boundaries. Particularly, we use the mean shift segmentation software—the EDISON System [16]—to obtain the initial segmentation map. Fig. 1a shows an example of the mean shift initial segmentation. For detailed information about mean shift and the EDISON system, please refer to [5–7,16]. In this paper, we only focus on the region merging.

2.1. Region representation and similarity measure

After mean shift initial segmentation, we have many small regions available. To guide the following region merging process, we need to represent these regions using some descriptor and define a rule for merging. A region can be described in many aspects, such as the color, edge [29], texture [30], shape and size of the region. Among them the color histogram is an effective descriptor to represent the object color feature statistics and it is widely used in pattern recognition [26] and object tracking [27], etc. In the context of region merging based segmentation, color histogram is more robust than the other feature descriptors. This is because the initially segmented small regions of the desired object often vary a lot in size and shape, while the colors of different regions from the same object will have high similarity. Therefore, we use the color histogram to represent each region in this paper.

The RGB color space is used to compute the color histogram in this paper. We uniformly quantize each color channel into 16 levels and then the histogram of each region is calculated in the feature space of $16\times16\times16=4096$ bins. Denote by Hist_R the normalized histogram of a region R. The next problem is how to merge the regions based on their color histograms so that the desired object can be extracted.

In the interactive image segmentation, the users will mark some regions as object and background regions. The key issue in region merging is how to determine the similarity between the unmarked regions with the marked regions so that the similar regions can be merged with some logic control. Therefore, we need to define a similarity measure $\rho(R,Q)$ between two regions R and Q to accommodate the comparison between various regions. There are some well-known goodness-of-fit statistical metrics such as the Euclidean distance, Bhattacharyya coefficient and the log-likelihood ratio statistic [25]. Here we choose to use the Bhattacharyya coefficient [1,25,27] to measure the similarity between R and Q

$$\rho(R,Q)=\sum_{u=1}^{4096}\sqrt{\mathrm{Hist}_R^u\cdot\mathrm{Hist}_Q^u} \tag{1}$$

where Hist_R and Hist_Q are the normalized histograms of R and Q, respectively, and the superscript u represents the uth element of them. Bhattacharyya coefficient ρ is a divergence-type measure which has a straightforward geometric interpretation. It is the cosine of the angle between the unit vectors

$$\left(\sqrt{\mathrm{Hist}_R^1},\ldots,\sqrt{\mathrm{Hist}_R^{4096}}\right)^\mathrm{T} \quad \text{and} \quad \left(\sqrt{\mathrm{Hist}_Q^1},\ldots,\sqrt{\mathrm{Hist}_Q^{4096}}\right)^\mathrm{T}$$

The higher the Bhattacharyya coefficient between R and Q is, the higher the similarity between them is.

The geometric explanation of the Bhattacharyya coefficient actually reflects the perceptual similarity between regions. If two regions have similar contents, their histograms will be very similar, and hence their Bhattacharyya coefficient will be very high, i.e. the angle between the two histogram vectors is very small. Certainly, it is possible that two perceptually very different regions may have very similar histograms. Fortunately, such cases are rare because the region histograms are local histograms and they reflect the local features of images. Even in case two perceptually different regions have similar histograms, the similarity between them is rarely the highest one in the neighborhood. Coupling with the "maximal similarity rule" introduced in Section 2.3, the Bhattacharyya similarity works well in the proposed region merging method.

The RGB/Bhattacharyya descriptor is a very simple yet efficient way to represent the regions and measure their similarity. It has been successfully used to measure the similarity between target model and candidate model in the popular kernel based object tracking method [27]. However, it should be stressed that other color spaces, such as the HSI color space, and other distance measures, such as the Euclidean distance between histogram vectors, can also be adopted in the proposed region merging scheme. In Section 3.3, we present examples by using HSI color space and Euclidean distance, respectively. The results are similar to those by using the RGB/Bhattacharyya descriptor.

2.2. Object and background marking

In the interactive image segmentation, the users need to specify the object and background conceptually. Similar to [10,13,17], the users can input interactive information by drawing markers, which could be lines, curves and strokes on the image. The regions that have pixels inside the object markers are thus called object marker regions, while the regions that have pixels inside the background markers are called background marker regions. Fig. 1b shows examples of the object and background markers by using simple lines. We use green markers to mark the object while using blue markers to represent the background. Please note that usually only a small portion of the object regions and background regions will be marked by the user. Actually, the less the required inputs by the users, the more convenient and more robust the interactive algorithm is.

After object marking, each region will be labeled as one of three kinds of regions: the marker object region, the marker background region and the non-marker region. To completely extract the object contour, we need to automatically assign each non-marker region with a correct label of either object region or background region. For the convenience of the following development, we denote by $\mathbf{M}_\text{O}$ and $\mathbf{M}_\text{B}$ the sets of marker object regions and marker background regions, respectively, and denote by $\mathbf{N}$ the set of non-marker regions.

2.3. Maximal similarity based merging rule

After object/background marking, it is still a challenging problem to extract accurately the object contour from the background because only a small portion of the object/background features are indicated by the user. The conventional region merging methods merge two adjacent regions whose similarity is above a preset threshold [14, Chapter 6.3]. These methods have difficulties in adaptive threshold selection. A big threshold will lead to incomplete merging of the regions belonging to the object, while a small threshold can easily cause over-merging, i.e. some object regions are merged into the background. Moreover, it is difficult to judge when the region merging process should stop.

Object and background markers provide some key features of object and background, respectively. Similar to graph cut and marker based watershed [4], where the marker is the seed and starting point of the algorithm, the proposed region merging method also starts from the initial marker regions and all the non-marker regions will be gradually labeled as either object region or background region. The lazy snapping cutout method proposed in [17], which combines graph cut with watershed based initial segmentation, is actually a region merging method. It is controlled by a max-flow algorithm [11]. In this paper, we present an adaptive maximal similarity based merging mechanism to identify all the non-marker regions under the guidance of object and background markers.

Let Q be an adjacent region of R and denote by $\bar{S}_Q = \{S_i^Q\}_{i=1,2,\ldots,q}$ the set of Q's adjacent regions. The similarity between Q and all its adjacent regions, i.e. $\rho(Q, S_i^Q)$, $i = 1,2,\ldots,q$, are calculated. Obviously, R is a member of $\bar{S}_Q$. If the similarity between R and Q is the maximal one among all the similarities $\rho(Q, S_i^Q)$, we will merge R and Q. The following merging rule is defined:

$$\text{Merge } R \text{ and } Q \quad \text{if } \rho(R,Q) = \max_{i=1,2,\ldots,q} \rho(Q, S_i^Q) \tag{2}$$

The merging rule (2) is very simple but it establishes the basis of the proposed region merging process. One important advantage of (2) is that it avoids the presetting of similarity threshold for merging control. Although "max" is an operator that is sensitive to outliers, we empirically found that it works well in our algorithm. This is mainly because that the histogram is a global descriptor of the local region and it is robust to noise and small variations. Meanwhile, the Bhattacharyya coefficient is the inner product of the two histogram vectors and it is also robust to noise and variations.

The marker regions cover only a small part of the object and background. Those object regions that are not marked by the user, i.e. the non-marker object regions, should be identified and not be merged with the background. Since they are from the same object, the non-marker object regions will usually have higher similarity with the marker object regions than the background regions. Therefore, in the automatic region merging process, the non-marker object regions will have high probabilities to be identified as object.

2.4. The merging process

The whole MSRM process can be divided into two stages, which are repeatedly executed until no new merging occurs. Our strategy is to merge background regions as many as possible while keep object regions from being merged. Once we merge all the background regions, it is equivalent to extracting the desired object. Some two-step strategies have been used in [22,23] for image pyramid construction. Different from [22,23], the proposed strategy aims for image segmentation and it is guided by the markers input by users.

In the first stage, we try to merge marker background regions with their adjacent regions. For each region $B \in \mathbf{M}_\text{B}$, we form the set of its adjacent regions $\bar{S}_B = \{A_i\}_{i=1,2,\ldots,r}$. Then for each A_i and $A_\text{i} \notin \mathbf{M}_\text{B}$, we form its set of adjacent regions $\bar{S}_{A_i} = \{S_j^{A_i}\}_{j=1,2,\ldots,k}$. It is obvious that $B \in \bar{S}_{A_i}$. The similarity between A_i and each element in $\bar{S}_{A_i}$, i.e. $\rho(A_i, S_j^{A_i})$, is calculated. If B and A_i satisfy the rule (2), i.e.

$$\rho(A_i, B) = \max_{j=1,2,\ldots,k} \rho(A_i, S_j^{A_i}) \tag{3}$$

then B and A_i are merged into one region and the new region will have the same label as region B:

$$B = B \cup A_i \tag{4}$$

Otherwise, B and A_i will not merge.

The above procedure is iteratively implemented. Note that in each iteration, the sets $\mathbf{M}_\text{B}$ and $\mathbf{N}$ will be updated. Specifically, $\mathbf{M}_\text{B}$ expands and $\mathbf{N}$ shrinks. The iteration stops when the entire marker background regions $\mathbf{M}_\text{B}$ will not find new merging regions.

After the region merging of this stage, some non-marker background regions will be merged with the corresponding background markers. However, there are still non-marker background regions which cannot be merged because they have higher similarity scores with each other than with the marker background regions. Fig. 2a shows that after the first stage merging, many regions belonging to the background (leaves, branches, etc.) are merged but there are still some non-marker background regions left.

To complete the task of target object extraction, in the second stage we will focus on the non-marker regions in $\mathbf{N}$ remained from the first stage. Part of $\mathbf{N}$ belongs to the background, while part of

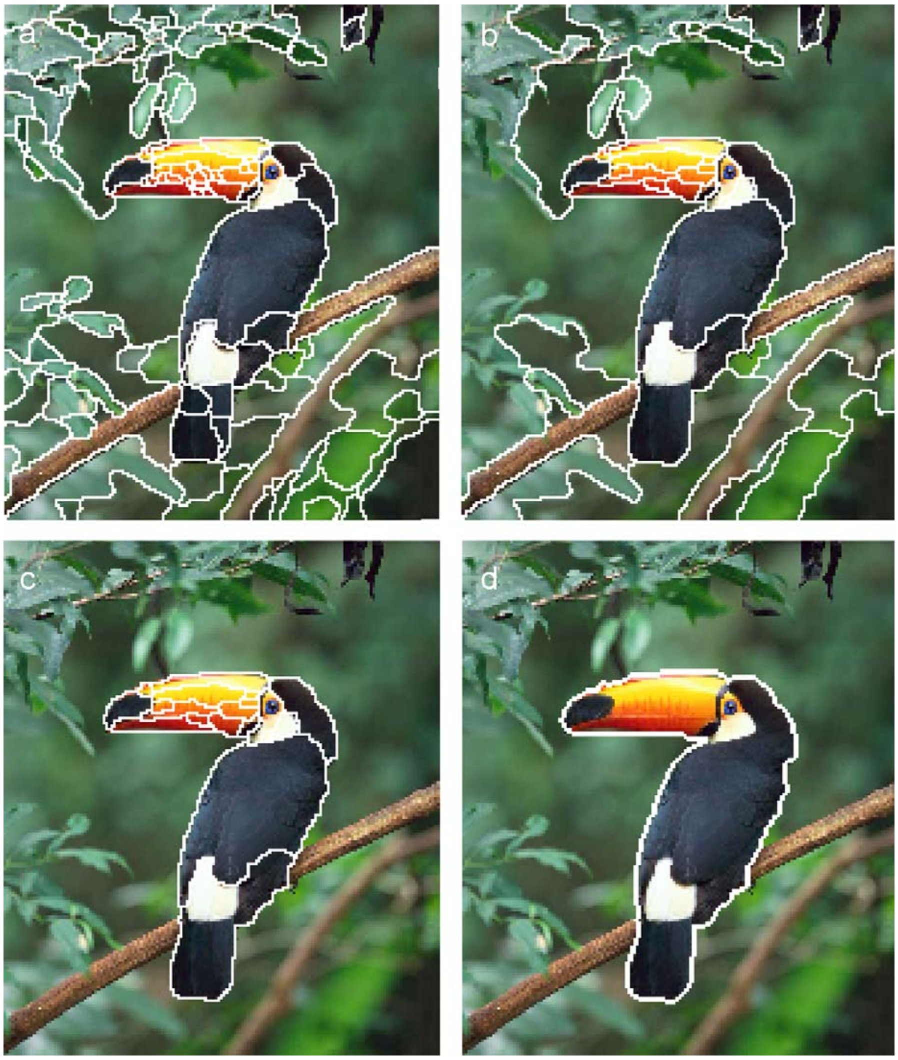

Fig. 2. Region merging process: (a) the first stage (1st round); (b) the second stage (1st round); (c) the first stage (2nd round); and (d) the merging results.

N belongs to the target object. In this stage, the non-marker object regions will be fused each other under the guidance of the maximal similarity rule and so do the non-marker background regions.

After the first stage, for each non-marker (background or object) region $P \in \mathbf{N}$, we form the set of its adjacent regions $\bar{S}_P=\{H_i\}_{i=1,2,\ldots,p}$. Then for each H_i that $H_i \notin \mathbf{M}_B$ and $H_i \notin \mathbf{M}_O$, we form its set of adjacent regions $\bar{S}_{H_i}=\{S_j^{H_i}\}_{j=1,2,\ldots,k}$. There is $P \in \bar{S}_{H_i}$. The similarity between H_i and each element in $\bar{S}_{H_i}$, i.e. $\rho(H_i,S_j^{H_i})$, is calculated. If P and H_i satisfy the rule (2), i.e.

$$\rho(P,H_i)=\max_{j=1,2,\ldots,k}\rho(H_i,S_j^{H_i}) \tag{5}$$

then P and H_i are merged into one region

$$P=P\cup H_i \tag{6}$$

Otherwise, P and H_i will not merge.

The above procedure is iteratively implemented and the iteration stops when the entire non-marker region set **N** will not find new merging regions. Fig. 2b shows the merging result after the second stage. We see that some non-marker background regions, as well as some non-marker object regions, are merged, respectively, in this stage.

The first and second stages of the algorithm are executed repeatedly until no new merging occurs. Fig. 2c shows the merging output of the first stage in the 2nd round. Since there is no more merging action, the algorithm stops here. In the end, each region is labeled as one of the two classes: object or background. Then we can easily extract the object contour by extracting only the object regions, as shown in Fig. 2d. In most of our experiments, the algorithm will end within 2–3 rounds. The whole algorithm can be summarized as follows:

The MSRM algorithm

```
Input: the initial mean shift segmentation result.
Output: the final segmentation map.

While there is region merging in the last loop
    Stage 1. Merging non-marker regions in $\mathbf{N}$ with marker
    background regions in $\mathbf{M}_B$
        Input: the initial segmentation result or the merging result
        of the second stage.
        (1-1) For each region $B \in \mathbf{M}_B$, form the set of its adjacent
        regions $\bar{S}_B=\{A_i\}_{i=1,2,\ldots,r}$.
        (1-2) For each $A_i$ and $A_i \notin \mathbf{M}_B$, form its set of adjacent
        regions $\bar{S}_{A_i} = \{S_j^{A_i}\}_{j=1,2,\ldots,k}$. There is $B \in \bar{S}_{A_i}$.
        (1-3) Calculate $\rho(A_i, S_j^{A_i})$. If $\rho(A_i, B) = \max_{j=1,2,\ldots,k}\rho(A_i, S_j^{A_i})$,
        then $B = B \cup A_i$. Otherwise, $B$ and $A_i$ will not merge.
        (1-4) Update $\mathbf{M}_B$ and $\mathbf{N}$ accordingly.
        (1-5) If the regions in $\mathbf{M}_B$ will not find new merging regions,
        the first stage ends. Otherwise, go back to (1-1).
    Stage 2. Merging non-marker regions in $\mathbf{N}$ adaptively
        Input: the merging result of the first stage.
        (2-1) For each region $P \in \mathbf{N}$, form the set of its adjacent
        regions $\bar{S}_P = \{H_i\}_{i=1,2,\ldots,p}$.
        (2-2) For each $H_i$ that $H_i \notin \mathbf{M}_B$ and $H_i \notin \mathbf{M}_O$, form its set of
        adjacent regions $\bar{S}_{H_i} = \{S_j^{H_i}\}_{j=1,2,\ldots,k}$. There is $P \in \bar{S}_{H_i}$.
        (2-3) Calculate $\rho(H_i, S_j^{H_i})$. If $\rho(P, H_i) = \max_{j=1,2,\ldots,k}\rho(H_i, S_j^{H_i})$,
        then $P = P \cup H_i$. Otherwise, $P$ and $H_i$ will not merge.
        (2-4) Update $\mathbf{N}$.
        (2-5) If the regions in $\mathbf{N}$ will not find new merging region,
        the second stage stops. Otherwise, go back to (2-1).
End
```

2.5. Convergence analysis

The proposed MSRM algorithm is an iterative method. It will progressively assign the non-marker background regions in $\mathbf{N}$ to $\mathbf{M}_B$, and then all the left regions in $\mathbf{N}$ are assigned to $\mathbf{M}_O$. It can be easily seen that the proposed method converge. We have the following theorem.

Theorem 1. *The MSRM algorithm in* Section 2.4 *converges, i.e. every region in* $\mathbf{N}$ *will be labeled as either object or background after a certain number of iterations.*

Proof. If a non-marker region $P \in \mathbf{N}$ has the maximal similarity (within its neighborhood) with a region in $B \in \mathbf{M}_B$, it will be merged with B, i.e. $B = P \cup B$, in the first stage of the proposed algorithm. If it has the maximal similarity with a region in $\mathbf{M}_O$, it will remain the same. If it has the maximal similarity with another non-marker region $P' \in \mathbf{N}$, P will be merged with P' in the second stage, i.e. $P = P \cup P'$. Then in the next round of iteration, P may be merged into $\mathbf{M}_B$, or it will continue merge with another P', or it will stay the same. If no non-marker region $P \in \mathbf{N}$ will be merged with a region in $\mathbf{M}_B$ or $\mathbf{N}$ after the zth round ($z > 1$), the algorithm will stop.

From the above analysis, we can see that the number of regions in N, denoted by n, will decrease in the process of iterative merging because some regions are labeled as background and some regions are merged with each other. Once n stops decreasing, the whole algorithm will stop and all the remaining regions in $\mathbf{N}$ will be labeled as object and merged into $\mathbf{M}_O$. Therefore, the proposed algorithm converges and it will label all the regions in $\mathbf{N}$. □

3. Experimental results

The proposed MSRM method is essentially an adaptive region merging method. With the markers input by the user, it will automatically merge regions and label the non-marker regions as object or background. In Section 3.1, we first evaluate the MSRM method qualitatively by several representative examples; in Section 3.2, we compare it quantitatively with the well-known graph cut algorithm; in Section 3.3, we test the MSRM under different color spaces, distance metrics and initial segmentation; at last in Section 3.4, we discuss the robustness of MSRM to user input markers as well as the failure cases of it.

3.1. Experimental analysis of the proposed method

Fig. 3 shows an example to extract the portrait (Mona Lisa) from a picture. After the initial segmentation of mean shift, the user inputs some interactive information: the green marker represents the object while the blue markers represent the background. Refer to Fig. 3a, the initial marker regions cover only part but representative features of

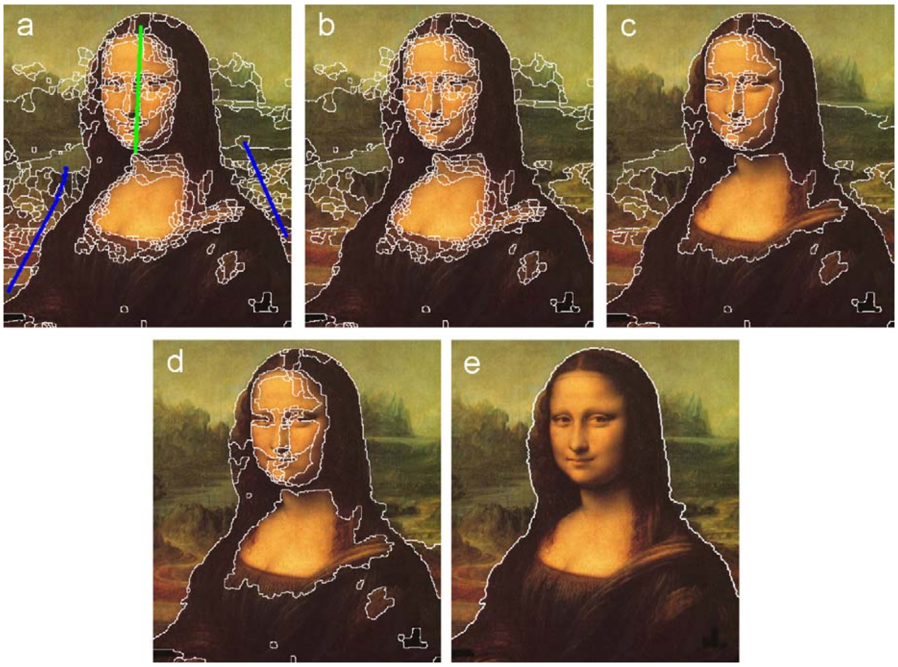

Fig. 3. Region merging process: (a) the initial mean shift segmentation results and the markers input by the user; (b) the first stage (1st round); (c) the second stage (1st round); (d) the first stage (2nd round); and (e) the extracted object contour.

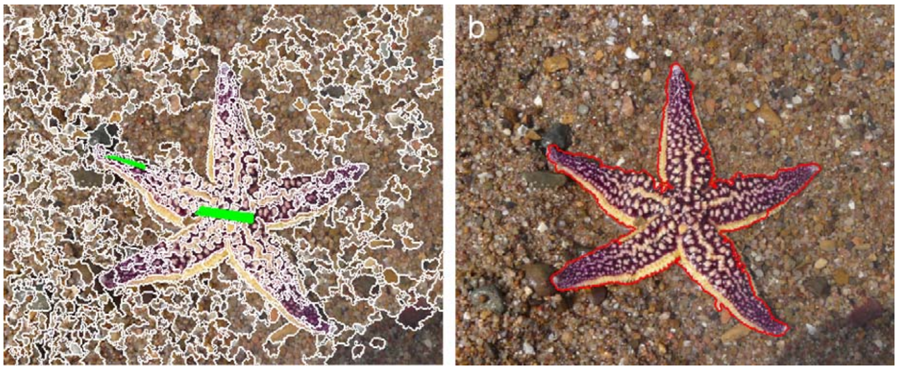

Fig. 4. (a) Initial mean shift segmentation and object markers and (b) the extracted object using the proposed method.

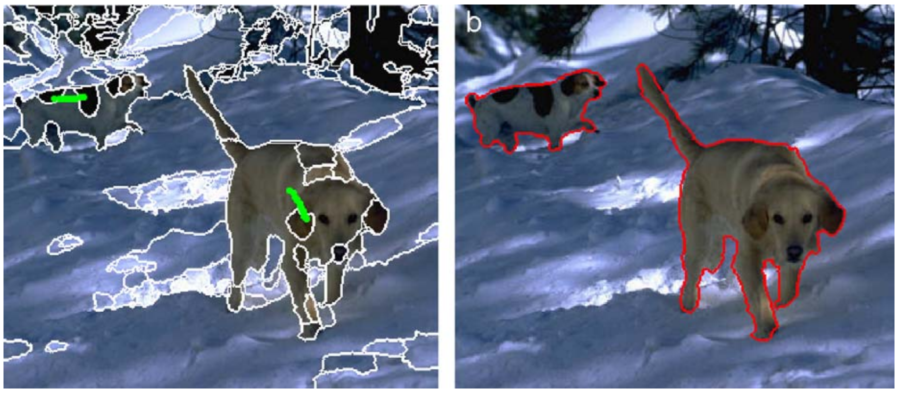

Fig. 5. Multiple object extraction: (a) initial mean shift segmentation and interactive information. The two green markers mark two objects. (b) The two extracted objects using the proposed method. For interpretation of the references to color in this figure legend, the reader is referred to the webversion of this article.

Table 1
The running time of MSRM on different images.

Image	Bird	Mona Lisa	Starfish-1	Dogs
Size of image	163×192	376×425	448×368	335×295
Number of regions after initial segmentation	170	522	1088	196
Running time (s)	6	32	80	12

the object and background regions. As shown in Figs. 3b–d, the object and background marker regions will propagate to all non-marker regions via iteratively implementing the two stage region merging process. Finally, Fig. 3e shows that the portrait is well extracted from the complex background.

In the second experiment, we want to separate a starfish from the complex background. Fig. 4a shows that the mean shift initial segmentation results in severe over segmentation for both the target object and background. In this image, since the starfish lies relatively in the center of the image, we implicitly specify the regions which locate in the border of the image as background markers. Therefore we only need to draw the object markers (green strokes) in the image. As shown in Fig. 4b, although there is no explicit user input background marker, the proposed MSRM method can still extract the desired object accurately.

The proposed MSRM scheme can be naturally extended to extract multiple objects. Fig. 5 shows an example to extract the two dogs in the snow background. Although the skin of the smaller dog in the left part of the scene is somewhat similar to the snow background, the proposed method still successfully separates it from the background. Meanwhile, although the contour of the bigger dog is complex, a simple marker was used to extract it out.

The execution time of the MSRM depends on a couple of factors, including the size of the image, the initial segmentation result, the user input markers and the content of the image. We implement the MSRM algorithm in the MATLAB 7.0 programming environment and run it on a PC with P4 2.6 GHz CPU and 1024 MB RAM. Table 1 lists the running time of the proposed method on the testing images bird, Mona Lisa, starfish-1 and dogs in Figs. 2–5, respectively.

3.2. Comparison with graph cut

In this section, we compare the MSRM algorithm with the well-known graph cut segmentation method [10–12] under the same user input markers. Since the original graph cut segmentation is a pixel based method, for a fair comparison with the proposed region based method, we extended the original pixel based graph cut (denoted by GC_P) to a region based graph cut (denote by GC_R), i.e. the nodes in the graph are mean shift segmented regions instead of the original pixels.

Fig. 6 shows the segmentation results of the three methods on eight test images. The first column shows the mean shift initial segmentation result and the input markers (for the last four images, the image boundary is set as the background marker); the second column shows the results by GC_P; the third column shows the results by GC_R; and the fourth column gives the results by MSRM. We can see that with the same user input markers, the proposed MSRM method achieves the best results, while GC_R performs better than GC_P. It can be seen that GC_R will miss some object regions and wrongly label some background regions as object regions.

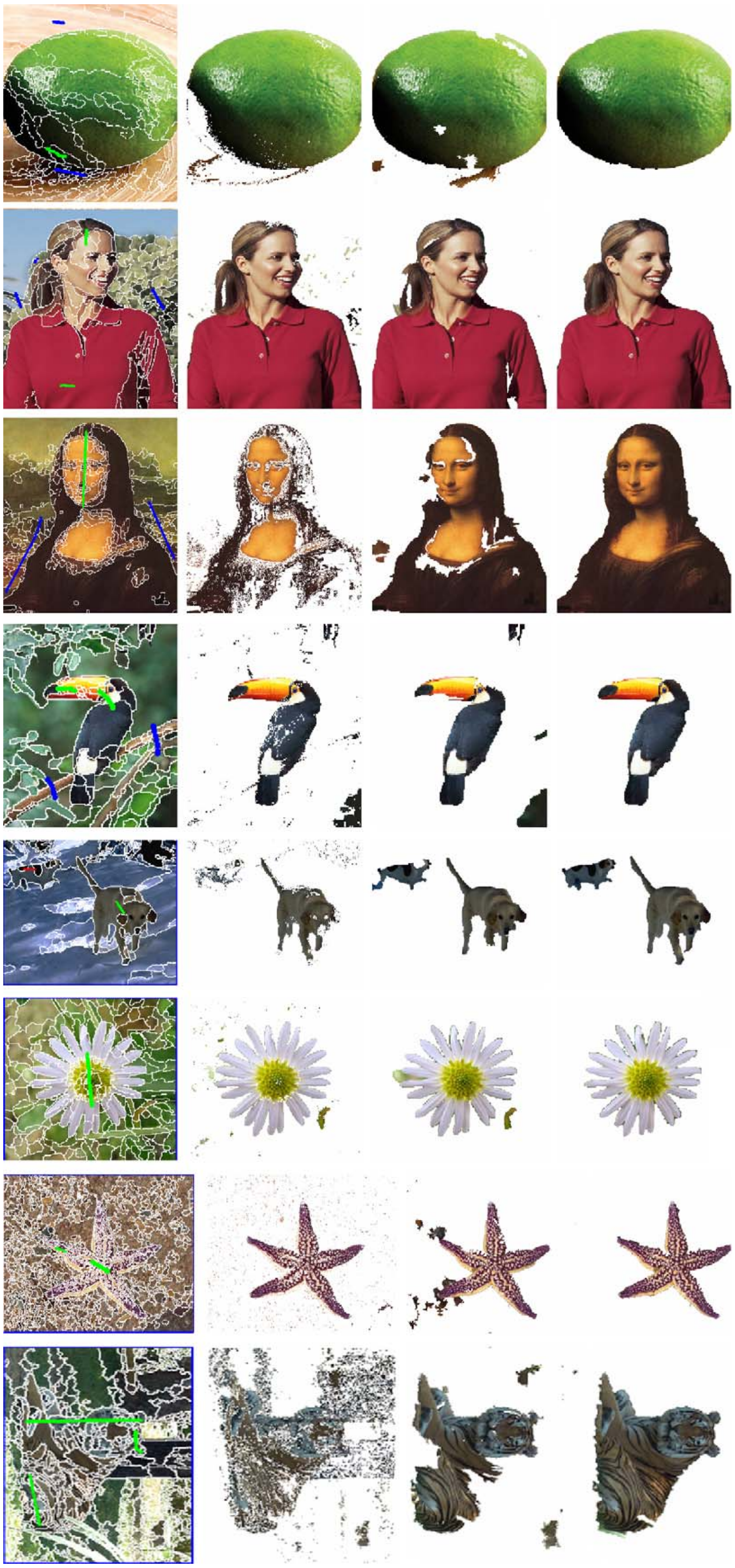

Fig. 6. Comparisons between the graph cut and proposed method. First column: initial segmentation and the input markers; second column: segmentation results by GCP; third column: segmentation results by GCR; last column: segmentation results by the proposed MSRM.

Table 2
The TPR and FPR values of different methods on the test images.

Image	Method	TPR (%)	FPR (%)
Fruit	GC_P	93.14	2.37
	GC_R	96.56	3.37
	MSRM	98.97	0.37
Woman	GC_P	97.58	2.99
	GC_R	96.82	0.73
	MSRM	98.53	0.44
Bird	GC_P	87.49	3.64
	GC_R	90.62	3.55
	MSRM	94.64	0.29
Dogs	GC_P	66.79	0.68
	GC_R	78.99	0.32
	MSRM	92.85	0.11
Mona Lisa	GC_P	54.08	2.02
	GC_R	90.71	2.34
	MSRM	98.85	0.71
Flower	GC_P	95.20	2.09
	GC_R	96.67	2.46
	MSRM	97.59	1.08
Tiger	GC_P	68.50	12.53
	GC_R	79.20	2.42
	MSRM	91.70	0.75
Starfish-1	GC_P	77.50	2.35
	GC_R	87.42	2.66
	MSRM	90.25	0.26

To quantitatively compare the three methods, we manually labeled the desired objects in the test images and took them as ground truth. Then we computed the true positive rate (TPR) and false positive rate (FPR) for these segmentation results. The TPR is defined as the ratio of the number of correctly classified object pixels to the number of total object pixels in the ground truth, and the FPR is defined as the ratio of the number of background pixels but classified as object pixels to the number of background pixels in the ground truth. Obviously, the higher the TPR is and the lower the FPR is, the better the method is. Table 2 lists the TPR and FPR results by the three comparison methods on the eight test images in Fig. 6. We can see that MSRM has the highest TPR and the lowest FPR simultaneously, which implies that it achieves the best segmentation performance. It can also be seen that GC_R has better performance than GC_P. This shows that by grouping the similar pixels into small homogenous regions, mean shift initial segmentation improve the robustness of graph cut to noise and small pixel variations.

3.3. MSRM under different color spaces, distance metrics and initial segmentation

Although the RGB color space and Bhattacharyya distance are used in the proposed MSRM method, other color spaces and distance metrics can also be used in MSRM. In this section, we present examples to verify the performance of MSRM under different color spaces and distance metrics, as well as different initial segmentation.

We first test the effect of color space on the region merging result. In this experiment, the RGB color images are converted into the HSI color space, and the HSI color histograms are then built. The

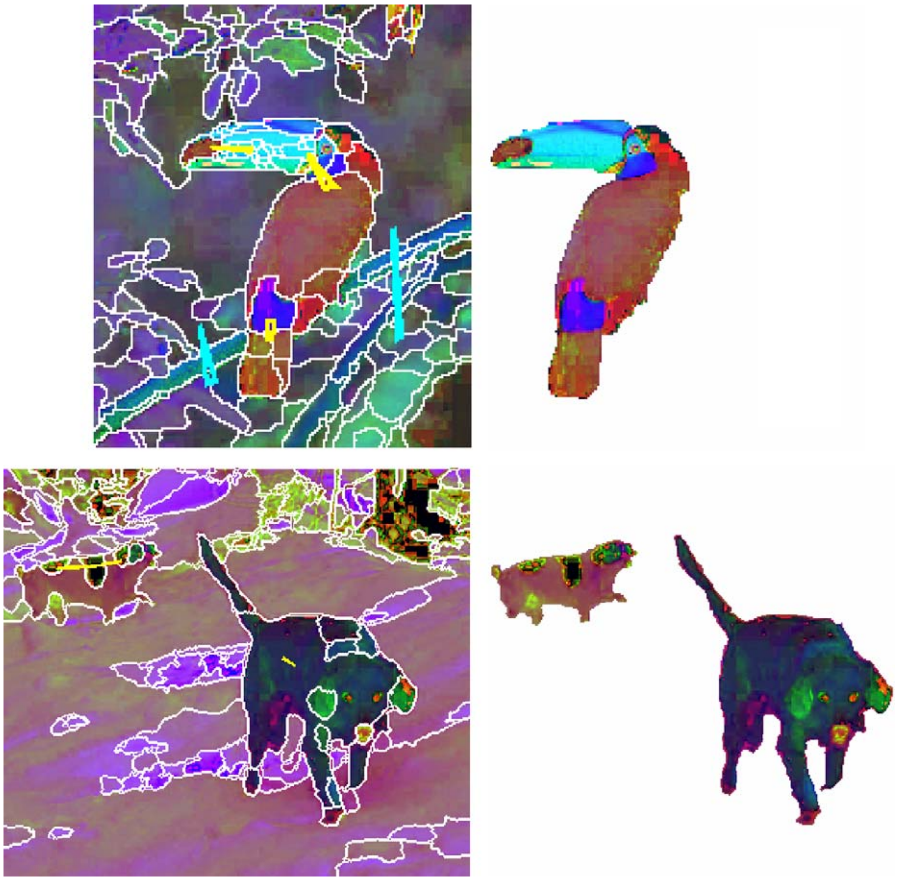

Fig. 7. Left column: initial segmentation by mean shift and the user input interactive information; right column: segmentation result by MSRM.

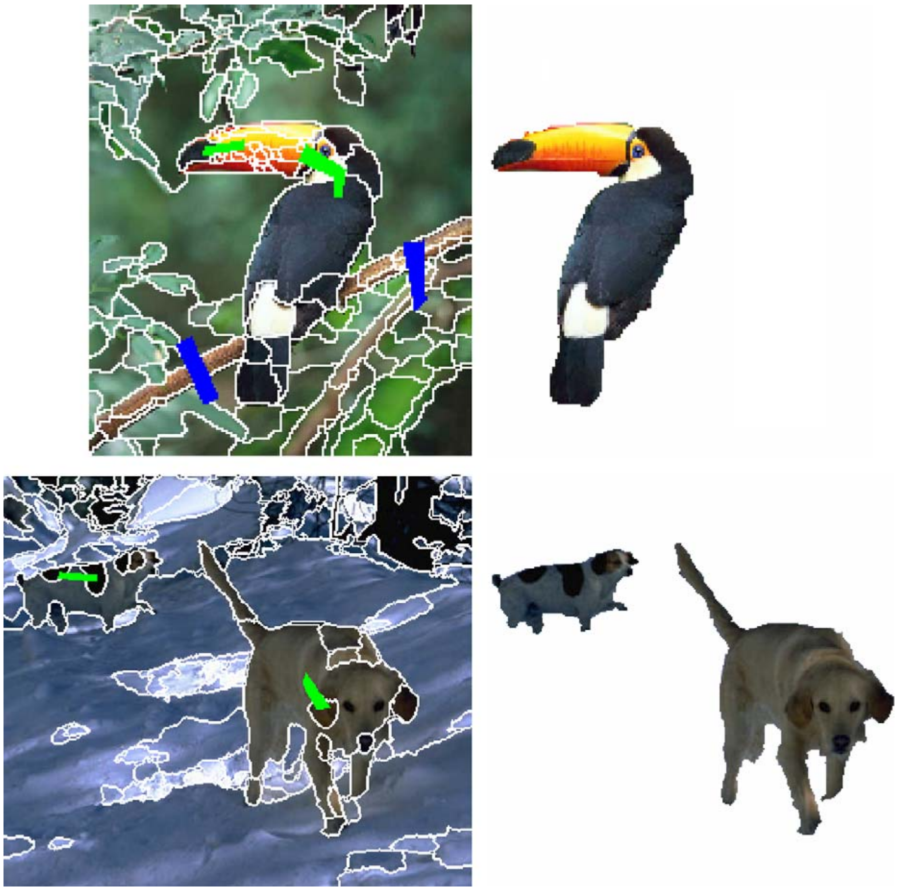

Fig. 8. Left column: initial segmentation by mean shift and the user input interactive information; right column: region merging result by using the Euclidean distance for similarity measurement.

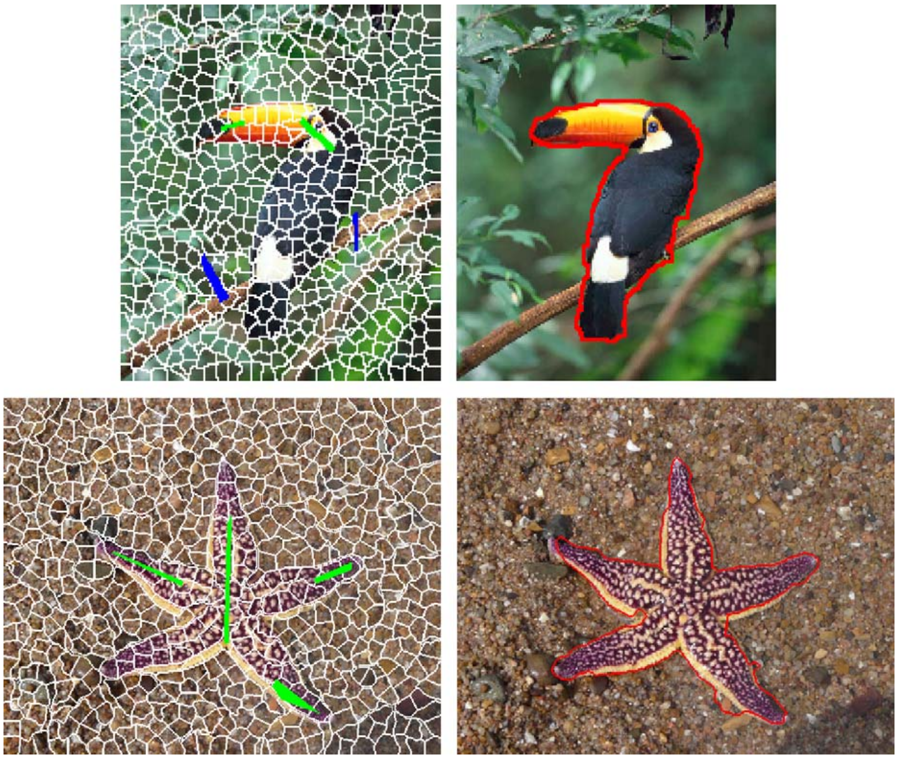

Fig. 9. Left column: initial segmentation by super-pixel method and the user input interactive information; right column: region merging result by the proposed MSRM method.

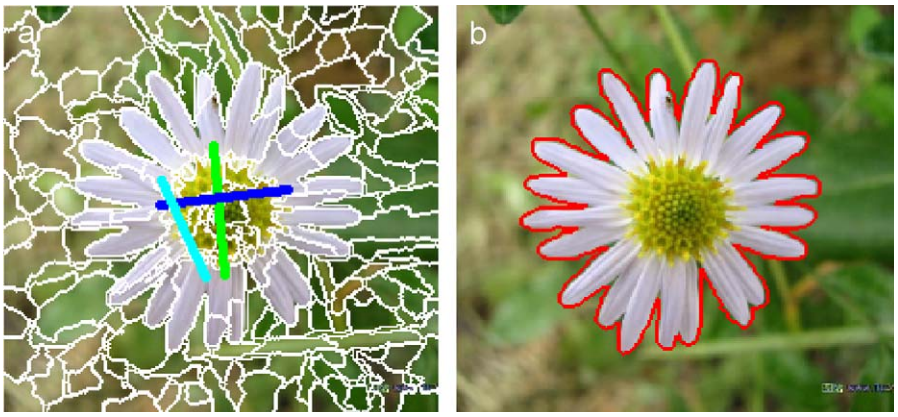

Fig. 10. (a) The initial mean shift segmentation and different markers (green, blue and cyan markers) input by the user; (b) the extracted object is the same under different user inputs using the proposed method. For interpretation of the references to color in this figure legend, the reader is referred to the webversion of this article.

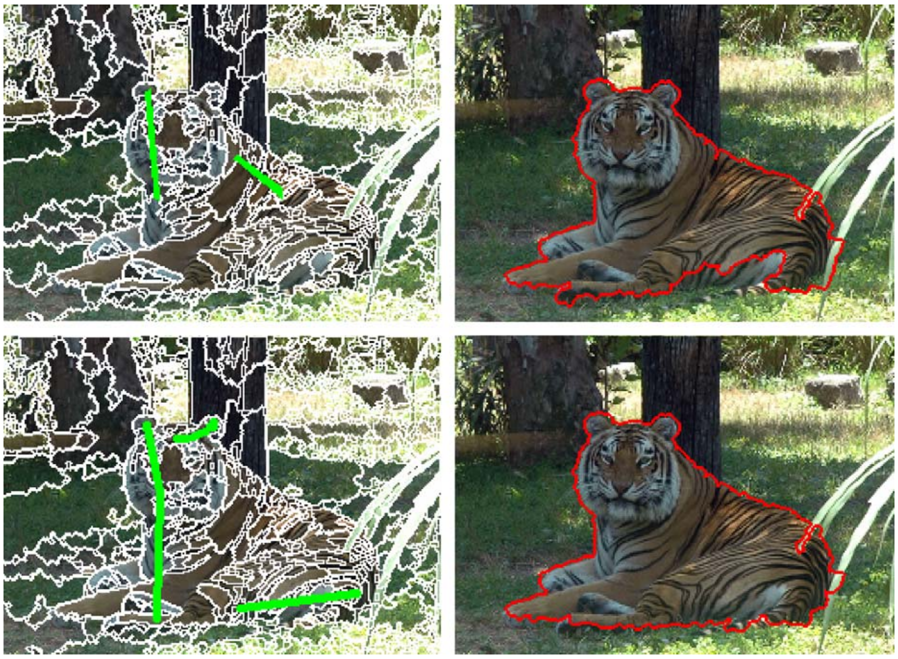

Fig. 11. Segment a tiger by the proposed MSRM method with two groups of markers.

Bhattacharyya coefficient is calculated by using the HSI color histograms as in (1) for similarity measurement. Fig. 7 shows the MSRM segmentation results on images bird and dogs. The left column shows the initially segmented images in the HSI color space, and the right column shows the finally segmented images by using the MSRM algorithm. We can see that the results are the same as those by using RGB color space with the Bhattacharyya distance.

We then test the effect of distance metric on the segmentation result. In this experiment, the RGB color space is used but we replace the Bhattacharyya distance by the Euclidean distance. Denote by Hist_R and Hist_Q the normalized RGB color histograms of two regions R and Q, the Euclidean distance between them is defined as

$$\rho(R,Q) = -\sqrt{\sum_{u=1}^{4096}(\text{Hist}_R^u - \text{Hist}_Q^u)^2} \tag{7}$$

Fig. 8 shows the segmentation results on images bird and dogs. We see that the results are the same as those by Bhattacharyya distance.

At last we test the MSRM algorithm with other initial segmentation. Besides mean shift, the super-pixel [28] is another popular initial segmentation method. Different from mean shift, it partitions evenly the image but into many small regions. In this experiment, the super-pixel method is used for initial segmentation. Fig. 9 shows the results on images bird and starfish-1. It can be seen that super-pixel leads to similar region merging results to those by mean shift. However, for some images, e.g. the starfish-1, it may require more user input markers. This is mainly because super-pixel has more over segmentation than mean shift, and hence the statistics of some regions segmented by super-pixel is not as robust as that by mean shift initial segmentation. For compensation, more markers may be required for the same result.

3.4. Robust analysis and failure cases

The proposed MSRM method is an interactive scheme, i.e. the users need to input markers. Therefore, the marker input by the user is important to segmentation. By our many experiments, we find

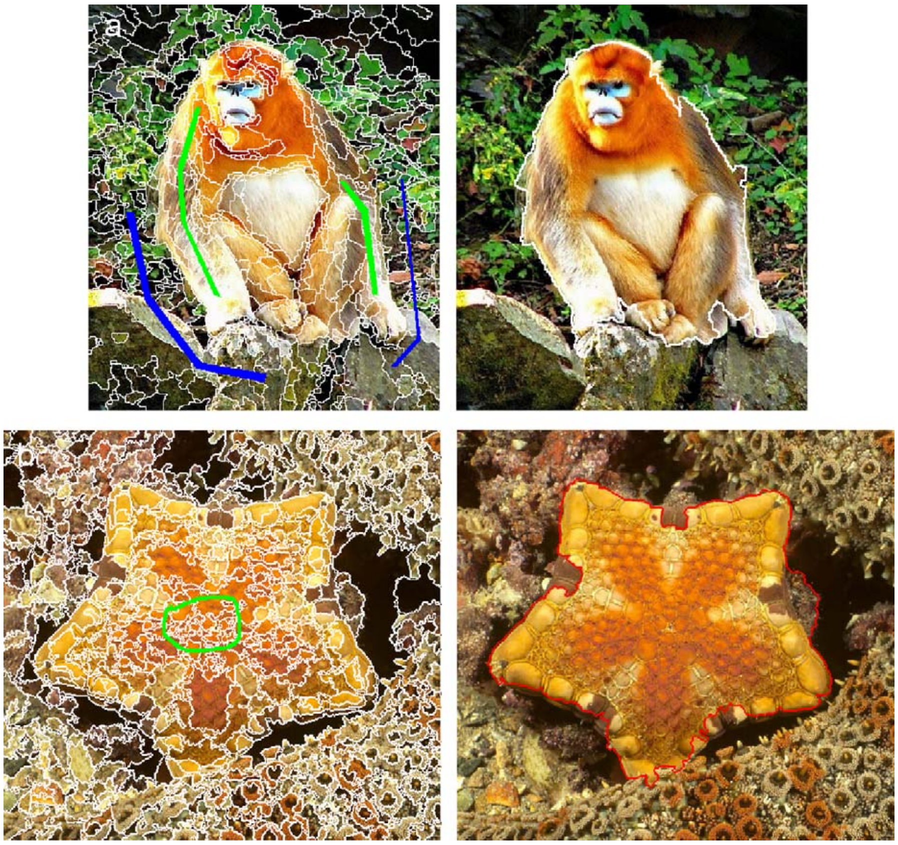

Fig. 12. Two failure examples of the proposed method: (a) parts of the object and background have very similar color features and (b) parts of the object are present in the background.

that the object can be correctly extracted as long as the markers can cover the main feature regions. To better illustrate this, we use an image with relatively simple features as an example. Refer to Fig. 10a, by using the three different (green, blue and cyan) object markers as user inputs, in Fig. 10b we can obtain the same object extraction result. (The regions in the border of the image are implicitly labeled as background markers.) This is because all the three object markers cover the main features (i.e. white and yellow colors) of the flower, i.e. the white petals and the yellow core.

In the experiment in Fig. 11, we try to separate a tiger from the complex background with two groups of markers. Obviously, the MSRM with more markers performs better than with few markers. Nonetheless, it still extracts the rough contour of tiger with even fewer markers. In general, the proposed MSRM algorithm could reliably extract the object contour from different backgrounds if the user input markers cover the main features of object and background. However, it may fail when shadow, low-contrast edges and ambiguous areas occur. For example, in Fig. 12a parts of the object regions are very similar to background. Although many markers were used to cover the object and background features, in some regions the proposed method does not achieve satisfying result. In Fig. 12b parts of the object are present in the background, so the final segmentation is not very good.

In addition, the proposed method is based on some initial segmentation such as mean shift or super-pixel. Therefore, if the initial segmentation does not provide a good basis to region merging, the proposed method may fail. Fortunately, many works [8,9,19] have been proposed or are under development to improve the mean shift segmentation, which will make the proposed method more robust and efficient in image segmentation tasks.

4. Conclusion

This paper proposed a novel region merging based interactive image segmentation method. The image is initially segmented by mean shift segmentation and the users only need to roughly indicate the main features of the object and background by using some strokes, which are called markers. Since the object regions will have high similarity to the marked object regions and so do the background regions, a novel maximal similarity based region merging mechanism was proposed to extract the object. The proposed scheme is simple yet powerful and it is image content adaptive. With the similarity based merging rule, a two stage iterative merging algorithm was presented to gradually label each non-marker region as either object or background. Extensive experiments were conducted to validate the proposed method in extracting single and multiple objects in complex scenes. The proposed scheme efficiently exploits the color similarity of the target object so that it is robust to the variations of input markers.

The proposed method provides a general region merging framework. It does not depend essentially on mean shift segmentation and other color image segmentation methods [3,9,15,16,28] can also be used for initial segmentation. Although some marker based interactive image segmentation methods (e.g. graph cut [10] and marker based watershed [4]) have been proposed, the proposed algorithm firstly exploits a novel adaptive maximal similarity based region merging mechanism. In the future, we will explore how to introduce pixel classification into the merging process to make the algorithm more intelligent.

References

[1] T. Kailath, The divergence and Bhattacharyya distance measures in signal selection, IEEE Transactions on Communications Technology 15 (1) (1967) 52–60.
[2] M. Kass, A. Witkin, D. Terzopoulos, Snake: active contour models, International Journal of Computer Vision 1 (4) (1987) 321–331.
[3] L. Vincent, P. Soille, Watersheds in digital spaces: an efficient algorithm based on immersion simulations, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (6) (1991) 583–598.
[4] F. Meyer, S. Beucher, Morphological segmentation, Journal of Visual Communication and Image Representation 1 (1) (1990) 21–46.
[5] Y. Cheng, Mean shift, mode seeking, and clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (8) (1995) 790–799.
[6] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5) (2002) 603–619.
[7] C. Christoudias, B. Georgescu, P. Meer, Synergism in low level vision, in: Proceedings of the International Conference on Pattern Recognition, vol. 4, 2002, pp. 150–155.
[8] Q. Luo, T.M. Khoshgoftaar, Efficient image segmentation by mean shift clustering and MDL-guided region merging, in: IEEE Proceedings of the International Conference on Tools with Artificial Intelligence, November 2004, pp. 337–343.
[9] J. Wang, B. Thiesson, Y. Xu, M.F. Cohen, Image and video segmentation by anisotropic Kernel mean shift, in: Proceedings of the European Conference on Computer Vision, Prague, Czech Republic, vol. 3022, 2004, pp. 238–249.
[10] P. Felzenszwalb, D. Huttenlocher, Efficient graph-based image segmentation, International Journal of Computer Vision 59 (2) (2004) 167–181.
[11] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (9) (2004) 1124–1137.
[12] V. Kolmogorov, R. Zabih, What energy functions can be minimized via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2) (2004) 147–159.
[13] Q. Yang, C. Wang, X. Tang, M. Chen, Z. Ye, Progressive cut: an image cutout algorithm that models user intentions, IEEE Multimedia 14 (3) (2007) 56–66.
[14] M. Sonka, V. Hlavac, R. Boyle, Image Processing, Analysis and Computer Vision, Thomson, 2007.
[15] B. Sumengen, Variational image segmentation and curve evolution on natural images, Ph.D. Thesis, University of California.
[16] EDISON software. ⟨http://www.caip.rutgers.edu/riul/research/code.html⟩.
[17] Y. Li, J. Sun, C. Tang, H. Shum, Lazy snapping, SIGGRAPH 23 (2004) 303–308.
[18] Y. Li, J. Sun, H. Shum, Video object cut and paste, SIGGRAPH 24 (2005) 595–600.
[19] S. Paris, F. Durand, A topological approach to hierarchical segmentation using mean shift, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[20] A. Levin, A. Rav-Acha, D. Lischinski, Spectral matting, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (10) (2008) 1699–1712.
[21] R. Carsten, K. Vladimir, B. Andrew, "Grabcut": interactive foreground extraction using iterated graph cuts, SIGGRAPH 23 (2004) 309–314.
[22] P. Meer, Stochastic image pyramids, Computer Vision, Graphics, and Image Processing (CVGIP) 45 (3) (1989) 269–294.
[23] J.M. Jolion, The adaptive pyramid: a framework for 2D image analysis, Computer Vision, Graphics, and Image Processing (CVGIP): Image Understanding 55 (3) (1992) 339–348.
[24] A. Blake, C. Rother, M. Brown, P. Perez, P. Torr, Interactive image segmentation using an adaptive GMMRF model, in: Proceedings of the European Conference on Computer Vision, 2004, pp. 428–441.
[25] K. Fukunaga, Introduction to Statistical Pattern Recognition, second ed., Academic Press, 1990.
[26] M.J. Swain, D.H. Ballard, Color indexing, International Journal of Computer Vision 7 (1) (2002) 11–32.
[27] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (5) (2003) 564–577.
[28] X. Ren, J. Malik, Learning a classification model for segmentation, ICCV03, vol. 1, pp. 10–17, Nice, 2003.
[29] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1998, pp. 232–237.
[30] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (7) (2002) 971–987.

About the Author—JIFENG NING received his College Diploma in Shenyang College of Technology in 1996 and Master of Engineering in Northwest A&F University in 2002. He is a lecturer with the College of Information Engineering, Northwest A&F University. Since 2004, he has been pursuing his Ph.D. degree in the State Key Laboratory of Integrated Service Networks, XIDIAN University. His research interests include computer vision, image segmentation and pattern recognition. He is now working as a research assistant in the Department of Computing, The Hong Kong Polytechnic University.

About the Author—LEI ZHANG received the B.S. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, PR China, the M.S. and Ph.D. degrees in Electrical and Engineering from Northwestern Polytechnical University, Xi'an, PR China, respectively, in 1998 and 2001. From 2001 to 2002, he was a research associate in the Department of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006 he worked as a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. Since January 2006, he has been an Assistant Professor in the Department of Computing, The Hong Kong Polytechnic University. His research interests include Image and Video Processing, Biometrics, Pattern Recognition, Computer Vision, Multisensor Data Fusion and Optimal Estimation Theory, etc.

About the Author—DAVID ZHANG graduated in Computer Science from Peking University in 1974 and received his M.Sc. and Ph.D. degrees in Computer Science and Engineering from the Harbin Institute of Technology (HIT), Harbin, PR China, in 1983 and 1985, respectively. He received the second Ph.D. degree in Electrical and Computer Engineering at the University of Waterloo, Waterloo, Canada, in 1994. From 1986 to 1988, he was a Postdoctoral Fellow at Tsinghua University, Beijing, China, and became an Associate Professor at Academia Sinica, Beijing, China. Currently, he is a Professor with the Hong Kong Polytechnic University, Hong Kong. He is Founder and Director of Biometrics Research Centers supported by the Government of the Hong Kong SAR (UGC/CRC). He is also Founder and Editor-in-Chief of the International Journal of Image and Graphics (IJIG), Book Editor, The Kluwer International Series on Biometrics, and an Associate Editor of several international journals. His research interests include automated biometrics-based authentication, pattern recognition, biometric technology and systems. As a principal investigator, he has finished many biometrics projects since 1980. So far, he has published over 200 papers and 10 books.

About the Author—CHENGKE WU received his B.Sc. in Wireless Communication in XIDIAN University in 1961. He is a professor with the School of Telecommunications Engineering and the State Key Laboratory of Integrated Service Networks, XIDIAN University. He was a visiting scholar in University of Pennsylvania, USA from 1980 to 1982, visiting professor in Nancy University, France, from 1990 to 1991, and visiting professor in The Chinese University of Hong Kong in 2000, 2001 and 2002, respectively. Professor Wu's research interests include image/video coding and transmission, multimedia, computer vision, etc. As a principle investigator, Professor Wu has successfully completed many projects, including the 863 High Technology Program of China and Natural Science Foundation of China (NSFC) Key Grant. He has won many awards, published four monographs and over 100 technical papers.