Pattern Recognition 52 (2016) 317–331

Contents lists available at ScienceDirect

Pattern Recognition

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

Robust single-object image segmentation based on salient transition region

Zuoyong Li [a,*], Guanghai Liu [b], David Zhang [c,d], Yong Xu [d]

[a] Department of Computer Science, Minjiang University, Fuzhou 350108, China
[b] School of Computer Science and Information Technology, Guangxi Normal University, Guilin 541004, China
[c] Biometrics Research Centre, Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
[d] Bio-Computing Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen 518055, China

ARTICLE INFO

Article history:
Received 27 April 2014
Received in revised form
8 October 2015
Accepted 11 October 2015
Available online 19 October 2015

Keywords:
Local complexity
Local variance
Salient transition region
Image thresholding
Image segmentation

ABSTRACT

Existing transition region-based image thresholding methods are unstable, and fail to achieve satisfactory segmentation accuracy on images with overlapping gray levels between object and background. This is because they only take the gray level mean of pixels in transition regions as the segmentation threshold of the whole image. To alleviate this issue, we proposed a robust hybrid single-object image segmentation method by exploiting salient transition region. Specifically, the proposed method first uses local complexity and local variance to identify transition regions of an image. Secondly, the transition region with the largest pixel number is chosen as salient transition region. Thirdly, a gray level interval is determined by using transition regions and image information, and one gray level of the interval is determined as the segmentation threshold by using the salient transition region. Finally, the image thresholding result is refined as final segmentation result by using the salient transition region to remove fake object regions. The proposed method has been extensively evaluated by experiments on 170 single-object real world images. Experimental results show that the proposed method achieves better segmentation accuracy and robustness than several types of image segmentation techniques, and enjoys its nature of simplicity and efficiency.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Image segmentation is a critical preprocessing step in image analysis and understanding. It aims to extract objects from an image by using image characteristics such as gray level, color, texture and location [1–3]. Image segmentation is used in a variety of applications such as biomedical image analysis [4], character identification [5] and target recognition [6]. Typically, image segmentation methods are divided into three categories, i.e, boundary-based methods [7–15], region-based methods [16–26] and hybrid methods [27–36].

Boundary-based image segmentation employs sharp variation of local image characteristics to find transitional image structures such as edges for separating object from background. Among this type of techniques, edge detection and graph cut are two representatives. There are some classical edge detectors such as Canny [8], Prewitt [9], Sobel and Roberts [10]. Canny [8] is the most widely used edge detector among them. However, Canny often suffers from edge fractures and undesirable object/background details as illustrated in Fig. 1(b). Graph cut-based methods [11–15] have attracted more and more attentions in recent years. They regard an image as a graph, and accordingly take an image segmentation problem as an iterative graph partition issue. Normalized cut (Ncut) [11] and isoperimetric cut (Isocut) [12] are two widely used methods. Fig. 1(c) and (d) shows segmentation results of both methods on the 10 images in the first group. Fig. 1(c) and (d) illustrates that Ncut obtains satisfactory segmentation results only on simple images with high contrast. In summary, Isocut usually obtains better segmentation accuracy than Ncut. However, Isocut still fails to achieve satisfactory segmentation accuracy on images with low contrast.

Region-based image segmentation defines uniformity of image region, and employs uniformity difference between object regions and background regions to perform segmentation. This type of technique includes region growth [18], region splitting and merging [19], clustering [20], image thresholding [21–25] and so on. Among them, global image thresholding is widely used due to the

* Corresponding author. Tel.: +86 13906926400; fax: +86 59183761607.
E-mail addresses: fzulzytdq@126.com, fzulzytdq@aliyun.com (Z. Li), liuguanghai009@163.com (G. Liu), csdzhang@comp.polyu.edu.hk (D. Zhang), laterfall@hitsz.edu.cn (Y. Xu).

http://dx.doi.org/10.1016/j.patcog.2015.10.009
0031-3203/© 2015 Elsevier Ltd. All rights reserved.

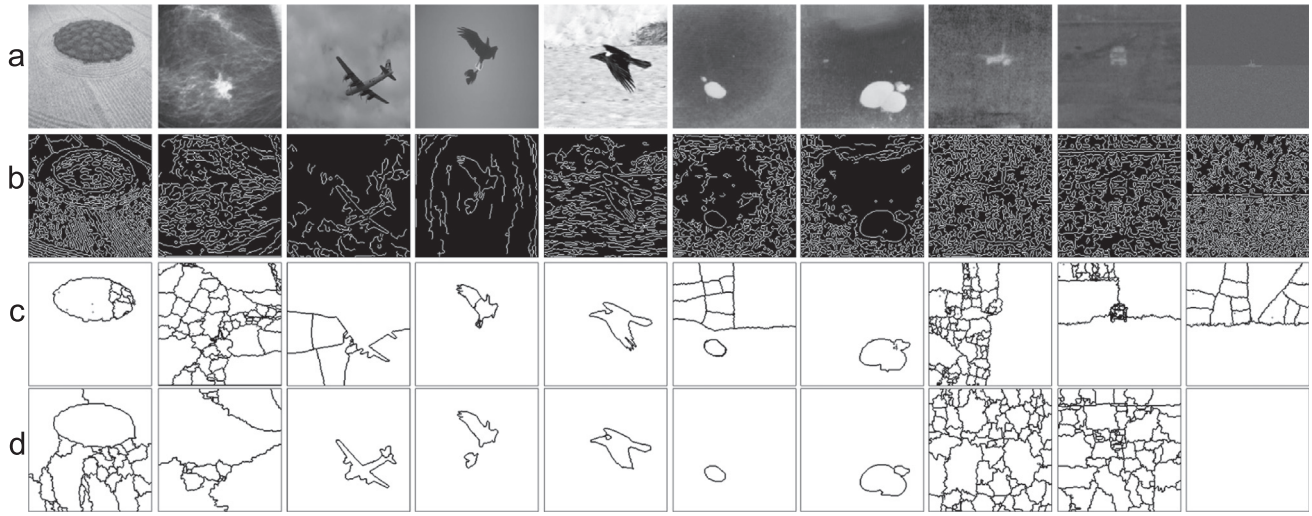

Fig. 1. Visual inspection of segmentation results obtained by applying three boundary-based methods to the 10 images in the first group, namely, Field, Mammography, Airplane1, Eagle, Crow, Airhole1, Airhole2, Airplane2, Car and Ship. (a) Original images. (b) Segmentation results of Canny [8]. (c) Segmentation results of Ncut [11] with stop criterion $=2\times10^{-2}$. (d) Segmentation results of Isocut [12] with stop criterion $=1\times10^{-5}$.

nature of simplicity and efficiency. Global image thresholding first chooses one of pixels' gray levels in an image as a segmentation threshold, and then performs image binarization by classifying image pixels with gray levels above the threshold as the object and the remainder as the background, or vice versa. Unfortunately, global thresholding fails to work on images with overlapping gray levels between object and background. Fig. 2(b) and (n) shows that perfect segmentations of some objects inevitably generate misclassifications of partial background pixels. Hence, single threshold in global image thresholding has its inherent limitation on this type of images. To remedy this limitation, many local/adaptive thresholding methods [25,26] were developed. For example, Niblack's method [25], which is dubbed Niblack, adaptively chooses segmentation threshold for each pixel according to the mean and standard deviation of its local patch. However, segmentation accuracy of Niblack is still unsatisfactory as illustrated in Fig. 2(e).

Hybrid image segmentation aims to obtain better results by simultaneously employing edge and region information. For example, a double threshold image binarization (DTIB) method [31] first uses Canny [8] to detect image edges, and finds a seed in each edge point's local neighborhood. Then, DTIB employs local image information to determine a high threshold and a low threshold, and obtains closed object contours by performing edge linking on the edge detection result. Finally, DTIB uses seed-growth technique integrating the object contours and the thresholding results obtained by the high and low thresholds to achieve final segmentation. Segmentation accuracy of DTIB relies heavily on the accuracy of Canny's edge detection as illustrated in Fig. 1(b) and Fig. 2(d). Transition region (TR)-based image thresholding [32–36] is a recently developed type of hybrid image segmentation technique. TR-based image thresholding first extracts image transition regions between object and background, and then takes gray level mean of pixels in the transition regions as segmentation threshold for image thresholding. TR-based image thresholding additionally considers image pixels' space information in comparison with conventional image thresholding, which makes it more reasonable and effective. However, it is essentially still a global image thresholding method. Hence, TR-based image thresholding inherits the above limitation of global image thresholding.

To this end, we are motivated to design a robust image binarization (RIB) method based on salient transition region. RIB has three contributions in comparison with existing TR-based image thresholding. Firstly, RIB proposes a new concept (i.e., salient transition region), and designs a criterion to measure saliency of transition regions. Secondly, RIB proposes a scheme of using both image transition regions and salient transition region to find a more appropriate segmentation threshold for perfectly segmenting object. Thirdly, RIB presents a scheme of refining image thresholding result by removing fake object regions which do not overlap with salient transition region, which significantly improves segmentation accuracy and robustness. We compare RIB with several types of image segmentation techniques, and experimental results on a series of single-object real world images show that RIB significantly improves segmentation accuracy and robustness, while being efficient thanks to its simplicity.

The rest of this paper is organized as follows. Section 2 introduces transition region based image thresholding, and analyzes its limitations. Section 3 describes the theory and implementation of the proposed method. Experimental results are reported in Section 4, and conclusions are drawn in Section 5.

2. Transition region-based image thresholding

The proposed method is developed to remedy the deficiencies of existing transition region (TR)-based image thresholding methods. This section will review the principle of TR-based image thresholding, and analyze its limitations.

A transition region is an image structure similar with the edge. An image transition region has three characteristics [32]. The first characteristic is that the transition region near an edge always exists, and usually has a width of several pixels. The second characteristic is that a true transition region should be located between the object and the background, and around the object. The last characteristic is that gray levels of pixels in the transition region usually change frequently and dramatically. In other words, transitional pixels usually have larger frequency and magnitude of gray level changes than non-transitional pixels, which provides key information for describing and extracting transition regions.

Existing TR-based image thresholding methods first extract transition regions of an image by using pre-designed transition region descriptor, and then take gray level mean of pixels in transition regions as segmentation threshold for image thresholding. TR-based image thresholding method belongs to global thresholding methods, and has two deficiencies. The first deficiency is that TR-based image

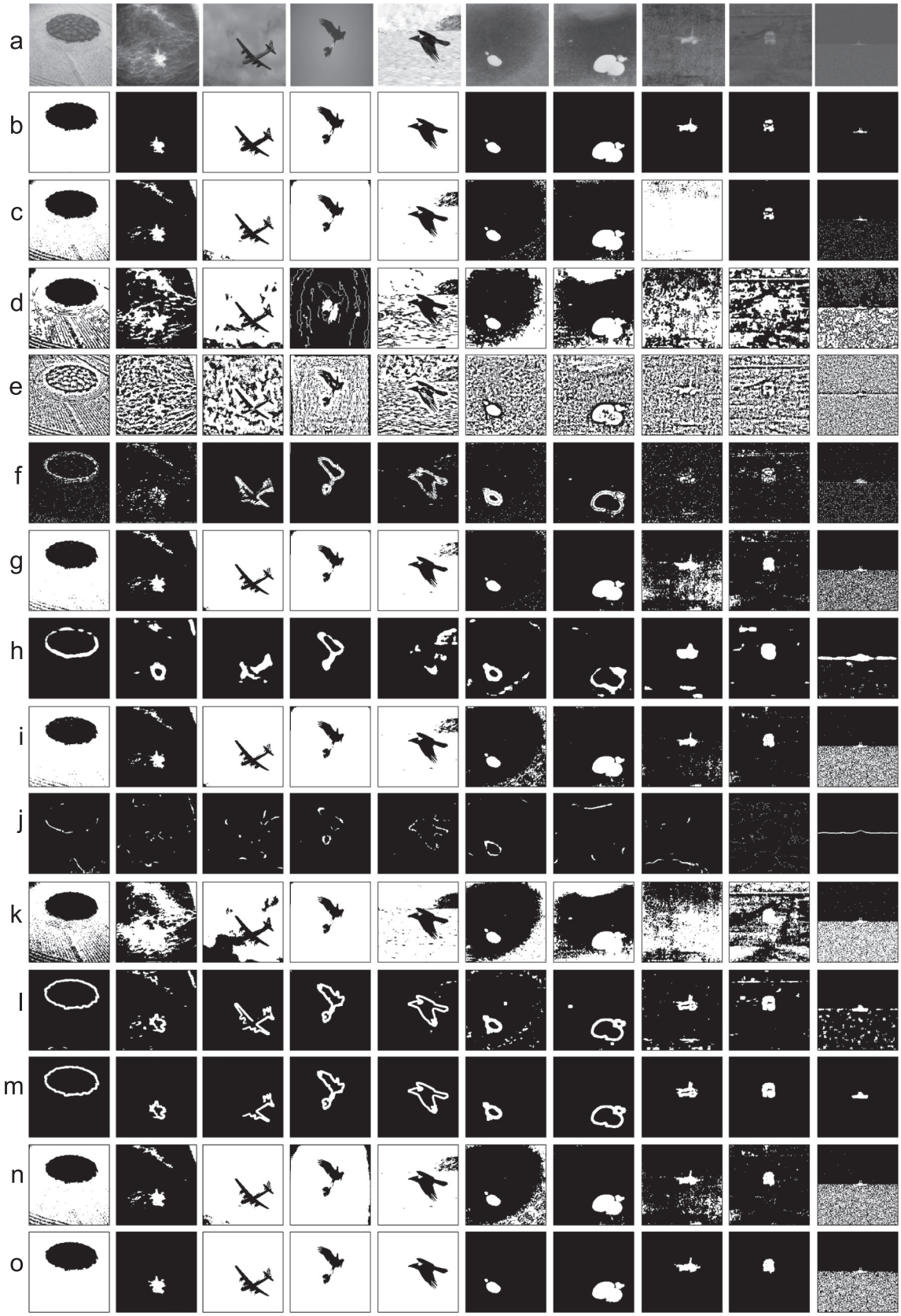

Fig. 2. Visual inspection of segmentation results on the 10 images in the first group. (a) Original images. (b) Manual ideal segmentation results (ground truths). (c) Segmentation results of PWT [23]. (d) Segmentation results of DTIB [31]. (e) Segmentation results of Niblack [25]. (f) Transition regions of LGLD [34]. (g) Segmentation results of LGLD [34]. (h) Transition regions of LE [33]. (i) Segmentation results of LE [33]. (j) Transition regions of MLE [35]. (k) Segmentation results of MLE [35]. (l) Transition regions of RIB. (m) Salient transition regions of RIB. (n) Image thresholding results of RIB. (o) Final segmentation results of RIB.

thresholding method cannot achieve satisfactory segmentation accuracy on images with overlapping gray levels between the object and the background. This is caused by the use of a single threshold in image thresholding. The gray level overlapping between the object and the background may be caused by uneven illuminations or image itself. Taking the 5th image in Fig. 2(a) as an example, we see that object pixels of the crow's body and background pixels on the top right corner overlap in gray levels. Perfect segmentation of the crow inevitably misclassifies those background pixels with overlapping gray levels. The second deficiency is that TR-based image thresholding method is unstable. As pointed out in [32], more accurate transition region should generate better segmentation result. However, results of Eagle (the 4th image) in Fig. 2(f)–(k) show that more accurate transition regions obtained by LGLD and LE generate worse segmentation results in comparison with MLE. The underlying reason is that TR-based image thresholding method simply takes gray level mean of pixels in transition regions as a global threshold, and does not employ transition regions adequately.

3. The proposed method

To remedy the deficiencies of TR-based image thresholding, a single-object image segmentation method based on salient transition region is proposed, the flow chart of which is shown in Fig. 3. The proposed method exploits image transition regions from a unique perspective. It significantly improves segmentation accuracy and robustness, and has the nature of simplicity and efficiency. Following sub-sections will describe the theory and implementation of the proposed method.

3.1. Transition region extraction

Transition region extraction aims to find transitional zone near a true object contour. Transitional pixels usually have more frequent and intensive gray level changes than non-transitional pixels. Based on this cue, many descriptors [32–36] have been developed to describe and extract transition regions in recent years. For example, local entropy (LE) method [33] extracts transition regions by measuring the frequency of gray level changes. Local gray level difference (LGLD) method [34] measures the magnitude of gray level changes. Modified local entropy (MLE) method [35] simultaneously measures both frequency and magnitude of gray level changes by integrating local complexity [36] with local variance. Local complexity has similar effect and simpler form on measuring the frequency of gray level changes in comparison with local entropy. Local variance, which is a common statistical measure reflecting magnitude of deviations between mean and individuals, is used to measure the magnitude of gray level changes in transition region description and extraction.

Without losing generality, for a given pixel $p(i,j)$ with an $m \times m$ local neighborhood Ω, its local complexity [36] is defined as

$$Lc(i,j) = C(\Omega) = \sum_{k=0}^{L-1} \mathrm{sgn}(k), \tag{1}$$

where

$$\mathrm{sgn}(k) = \begin{cases} 1 & \text{if } \exists\, f(x,y) = k, \\ 0 & \text{otherwise}, \end{cases} \tag{2}$$

(x, y) denotes local coordinate in the neighborhood Ω, and the gray level interval of the whole image is $[0\ L-1]$. Local variance [35] of $p(i,j)$ is defined as

$$Lv(i,j) = \sigma^2(\Omega) = \frac{1}{\mathrm{m}^2-1} \sum_{\mathrm{x}=1}^{\mathrm{m}} \sum_{y=1}^{\mathrm{m}} (\mathrm{f}(x,y) - \bar{\mathrm{f}})^2, \tag{3}$$

where (x, y) also denotes local coordinate in Ω, $f(x, y)$ denotes the gray level of $p(x, y)$, and $\bar{f}$ denotes the gray level mean of Ω.

When Ω is moved pixel by pixel within the image from left to right and top to bottom, local complexity and local variance of each pixel can be calculated. Note that, in order to apply Ω to the whole image, four boundaries of each image are extended through mirror-reflecting by enough pixels. Local complexities and local variances of all pixels in an image respectively compose two image matrices

$$Lc = \begin{bmatrix} Lc(1,1) & Lc(1,2) & \ldots\ldots & Lc(1,n_w) \\ Lc(2,1) & Lc(2,2) & \ldots\ldots & Lc(2,n_w) \\ \ldots & \ldots & \ldots\ldots & \ldots \\ Lc(n_h,1) & Lc(n_h,2) & \ldots\ldots & Lc(n_h,n_w) \end{bmatrix}, \tag{4}$$

$$Lv = \begin{bmatrix} Lv(1,1) & Lv(1,2) & \ldots\ldots & Lv(1,n_w) \\ Lv(2,1) & Lv(2,2) & \ldots\ldots & Lv(2,n_w) \\ \ldots & \ldots & \ldots\ldots & \ldots \\ Lv(n_h,1) & Lv(n_h,2) & \ldots\ldots & Lv(n_h,n_w) \end{bmatrix}, \tag{5}$$

where n_h and n_w denote the height and width of the image, respectively. Then, both image matrices are normalized [35] to prevent one of them from being neglected owing to large difference between their values in the following way:

$$NLc(i,j) = \frac{Lc(i,j) - \min\limits_{\forall(x,y)} Lc(x,y)}{\max\limits_{\forall(x,y)} Lc(x,y) - \min\limits_{\forall(x,y)} Lc(x,y)}, \tag{6}$$

$$NLv(i,j) = \frac{Lv(i,j) - \min\limits_{\forall(x,y)} Lv(x,y)}{\max\limits_{\forall(x,y)} Lv(x,y) - \min\limits_{\forall(x,y)} Lv(x,y)}. \tag{7}$$

Finally, both normalized local complexity and normalized local variance are synthesized [35] as the transition region descriptor *NLCV*, where

$$NLCV(i,j) = w \times NLc(i,j) + (1-w) \times NLv(i,j), \tag{8}$$

and the weight w between 0 and 1 is used to balance contributions of normalized local complexity and normalized local variance. When $w=1$, *NLCV* degenerates to local complexity. On the contrary, when $w=0$, *NLCV* degenerates to local variance. *NLCV* values of all pixels construct a transition region description matrix *NLCV*.

Based on the matrix *NLCV*, image transition regions can be extracted via the following steps:

(1) Convert *NLCV* into a vector *SNLCV*, and sort all elements of *SNLCV* in descending order.
(2) Choose αn pixels corresponding to the first αn elements in *SNLCV* as transitional pixels and compose transition regions, where α is a parameter and n denotes the total pixel number in the image. The principle of transition region extraction is that transitional pixels should have larger local complexities and local variances owing to frequent and intensive gray level changes in local neighborhoods than non-transitional pixels. To denote transitional pixels, we define a label matrix *TR* as

$$TR(i,j) = \begin{cases} 1 & \text{if } p(i,j) \text{ is a transitional pixel}, \\ 0 & \text{otherwise}. \end{cases} \tag{9}$$

3.2. Salient transition region extraction

The goal of salient transition region extraction is to choose one true and representative (i.e., salient) transition region from all transition regions, which probably contain fake transition regions intra the object or the background. The extracted salient transition region will be used to search for an appropriate segmentation

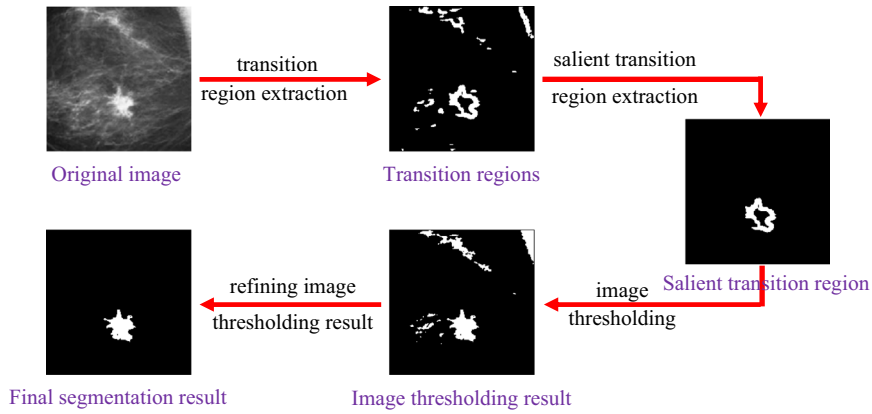

Fig. 3. Flow chart of the proposed method.

threshold for image thresholding in the next section. Hence, the salient transition region should be a true transition region between the object and the background. In other words, the salient transition region should be composed of partial object pixels and partial background pixels. Analyzing gray levels of pixels in the salient transition region can help us find the optimal segmentation threshold to distinguish the object from the background. Therefore, for extracting of salient transition region, the most important rule is to ensure that the extracted salient transition region is true transition region between the object and the background. Secondly, a salient transition region with more pixels provides more gray level information for determining the optimal segmentation threshold. Based on both rules, a simple and effective measure (i.e., pixel number) is used to quantify the saliency of transition regions. The process of salient transition region extraction is as follows:

(1) Rectify transition regions located at any of the four image boundaries. That is, elements of the transition region label matrix *TR* corresponding to those regions are set to zeros. This rectification aims to remove undesirable transition regions caused by some factors such as uneven illuminations at image boundaries, and to prevent any of them from being chosen as salient transition region. As we know, an image object is usually not located at image boundaries, and salient transition region around the object is usually not located at image boundaries accordingly. Even though the object is present at image boundaries, and a true transition region, which is located at image boundaries and has the largest pixel number, is not chosen as salient transition region due to the above removal during our transition region rectification process. We can choose the transition region with the second largest pixel number as salient transition region. The chosen salient transition region is usually also a true transition region. Both selections of salient transition region have little effect on subsequent steps. This is because salient transition region does not necessarily surround the whole object, which is illustrated in Fig. 4.
(2) Use different numbers to label all transition regions via Matlab function "bwlabel", where each connected transition region has a unique label number.
(3) Calculate the number of pixels in each connected transition region.
(4) Take the transition region with the largest pixel number as salient transition region. If there are two or more transition regions with the same largest pixel number, only one of them is chosen as salient transition region. Here, the proposed method adopts a simplest scheme to extract salient transition region, that is, it chooses the transition region with the smallest label number as salient transition region. To denote salient transition region, we define another label matrix *STR* as

$$STR(i,j) = \begin{cases} 1 & \text{if } p(i,j) \text{ is in salient transition region,} \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

In fact, the probability of two or more transition regions having the same largest pixel number is very low. To validate this hypothesis, we specifically recorded the number of transition regions with the largest pixel number for each image used in our experiments. Experimental results show that only one image (the 112th image) in the second group has two transition regions with the same largest pixel number (348), and the other 169 images have only one transition region with the largest pixel number. The special sample is shown in Fig. 4(a). The image has three dark boundaries as pointed out by the red arrows, which makes those boundary pixels have large local variances due to large gray level differences between them and their neighboring background pixels. Those boundary pixels are bright pixels in Fig. 4(b), showing large probability of being extracted as transition region. To better display those boundary pixels, we specifically added black frames on Fig. 4(b)–(d). Fig. 4(c) illustrates that those boundary pixels are extracted as transition region. Fig. 4(d) demonstrates that the rectification in the first step of salient transition region extraction successfully removes the fake transition region located at image boundaries.

After performing the transition region rectification, there are two transition regions with the same largest pixel number (348) shown in Fig. 4(e1) and (e2). Both transition regions are respectively chosen as salient transition region and exhibited with manual segmentation result by color images Fig. 4(f1) and (f2), where salient transition regions are colored with cyan. Each color image is obtained by applying Matlab function "imagesc" to a three value image matrix, where 0, 255 and 128 denote object pixel value, background pixel value and salient transition region pixel value, respectively. Fig. 4 (f1) and (f2) shows that both salient transition regions are composed of partial object pixels and partial background pixels, and both of them are true transition regions. Fig. 4(g1)–(g2) and (h1)–(h2) show that two salient transition regions respectively find similar segmentation thresholds (i.e., $T^*=156$ and $T^*=157$), and obtain similar final segmentation results (i.e., ME=0.0065582 and ME=0.0065155), where ME (misclassification error) is defined in Eq. (23). In summary, when there are two or more transition regions with the same largest pixel number, choosing either one as salient transition region from them has little effect on final segmentation result. In this case, the proposed method adopts a simplest scheme to extract salient

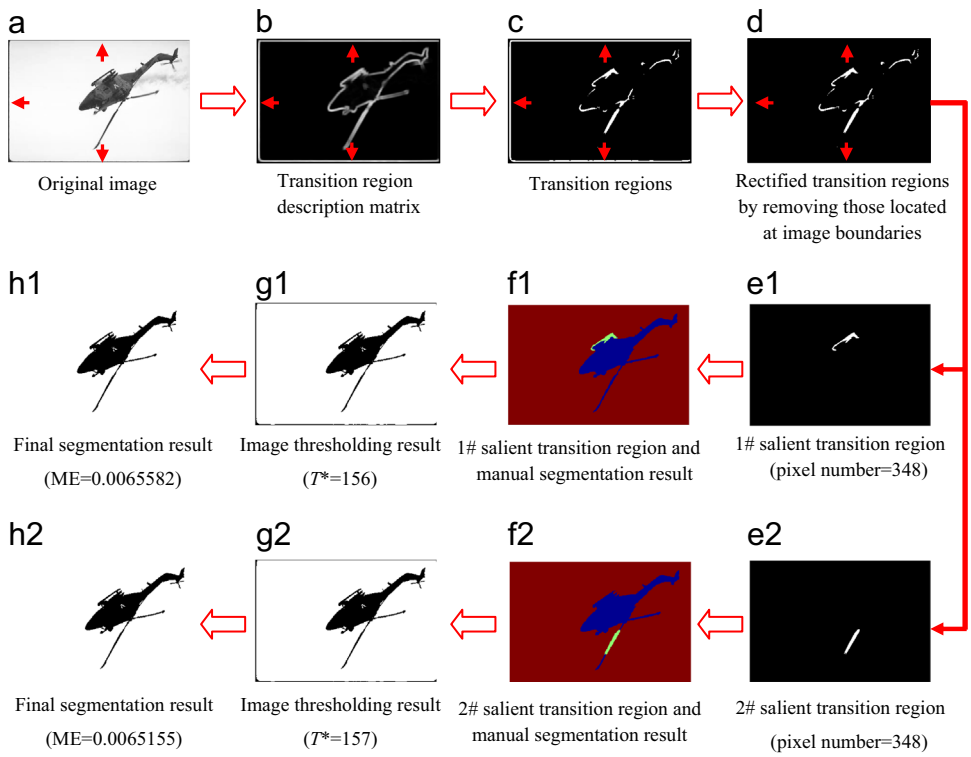

Fig. 4. An example containing two possible salient transition regions, where T^* denotes segmentation threshold defined in Eq. (21) and ME denotes misclassification error measure defined in Eq. (23).

transition region, i.e., by choosing transition region with the smallest label number among them as salient transition region.

3.3. Image thresholding

This step first searches for a more appropriate segmentation threshold than the gray level mean of pixels in transition regions, and then obtains initial segmentation result by image thresholding. In this section, a novel scheme is proposed to determine the segmentation threshold. The proposed scheme first uses global image information and transition regions to find a gray level interval as the candidate threshold interval, and then employs salient transition region to choose one gray level of the interval as the segmentation threshold for image thresholding.

Two considerations rise in the research of the candidate threshold interval. One consideration is that object in a grayscale image can be identified, as it differs from its surrounding background in gray levels. The other consideration is that transition regions have more frequent and intensive gray level changes than non-transition regions, which makes transition regions attract more attentions in human visual perception and helps human eye distinguish the object from the background. True transition regions should be located between the object and the background. In other words, true transition regions should be composed of partial object pixels and partial background pixels. Hence, the optimal segmentation threshold should have a strong relationship with true transition regions. This cue suggests us search for the optimal segmentation threshold from a gray level interval around gray level mean of pixels in transition regions, because most of the transition regions are true transition regions although there may be fake transition regions intra object or background.

The gray level interval (i.e., the candidate threshold interval) is determined by the following steps.

(1) Calculate gray level mean, T_m, of all pixels in transition regions via the following formula,

$$T_m = \frac{\sum_i \sum_j (TR(i,j) \times f(i,j))}{\sum_i \sum_j TR(i,j).} \tag{11}$$

(2) Calculate two gray levels, t_1 and t_2, via the following formulae,

$$t_1 = T_m - \beta \times \sigma, \tag{12}$$

$$t_2 = T_m + \beta \times \sigma, \tag{13}$$

where σ denotes the standard deviation of gray levels in the whole image, and β is a parameter.

(3) Define the candidate threshold interval R as

$$R = [t_1 \;\; t_2] \cap [0 \;\; 255]. \tag{14}$$

Once the interval R is determined, an appropriate segmentation threshold can be chosen from R via the following steps.

(1) For a given gray level t in R, its image thresholding (binarization) result B_t is obtained using

$$B_t(i,j) = \begin{cases} 1 & \text{if } f(i,j) > t, \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

(2) Perform a reversion operation on B_t using the following criterion

$$B_t = \begin{cases} \overline{B}_t & \text{if } n_o > n_b, \\ B_t & \text{otherwise,} \end{cases} \tag{16}$$

where

$$\overline{B}_t(i,j) = \begin{cases} 0 & \text{if } B_t(i,j) = 1, \\ 1 & \text{if } B_t(i,j) = 0, \end{cases} \tag{17}$$

$$n_o = \sum_i \sum_j B_t(i,j), \tag{18}$$

$$n_b = n - n_o, \tag{19}$$

and n is the total number of image pixels. The reversion operation aims to guarantee that 1 and 0 denote object pixels and background pixels in B_t, respectively. Here, it is assumed that the size of object is smaller than that of the background in an image. The assumption is usually suitable for most of natural images.

(3) Calculate the number of intersected pixels between the object in B_t and the salient transition region STR using

$$N(t) = \sum_i \sum_j (B_t(i,j) \times STR(i,j)). \tag{20}$$

(4) Define the segmentation threshold T^* as

$$T^* = \underset{t \in R}{\mathrm{Argmax}}\{N(t)\}. \tag{21}$$

If t meeting with Eq. (21) is not sole, only the smallest t is taken as T^*.

To accurately segment the object, we find T^* by maximizing the number of intersected pixels between the object in image thresholding result B_t and the salient transition region STR as formulated by Eqs. (20) and (21).

As described in Section 3.2, salient transition region should be true transition region between object and background. In other words, salient transition region should be composed of partial object pixels and partial background pixels. Maximizing the number of intersected pixels between the object in image thresholding result and salient transition region indeed obtain favorable object segmentation. To explain the underlying reason, we synthesized a single-object image shown in Fig. 5(a), where the circle colored with cyan denotes the object and the ring fragment colored with blue denotes salient transition region. Fig. 5(b)–(d) shows segmented objects in image thresholding results of three different thresholds by three red circles on the basis of Fig. 5(a). From Fig. 5(b)–(d), the number of intersected pixels between the object in image thresholding result and the salient transition region increases, meanwhile image thresholding result segments the object more accurately. Hence, when determining the segmentation threshold T^*, we maximize the number of intersected pixels between the object in image thresholding result and the salient transition region. In addition, Fig. 6 shows that image thresholding results obtained by T^* obtain more accurate objects in comparison with those obtained by T_m.

3.4. Refining image thresholding result

Image thresholding result obtained by the threshold T^* may misclassify partial background pixels into object pixels due to the overlapping gray levels between them as illustrated in Fig. 6(c). This section presents a salient transition region-based scheme to remove fake object regions composed of misclassified background pixels, and to refine image thresholding result obtained by T^*. The detailed refining process is as follows.

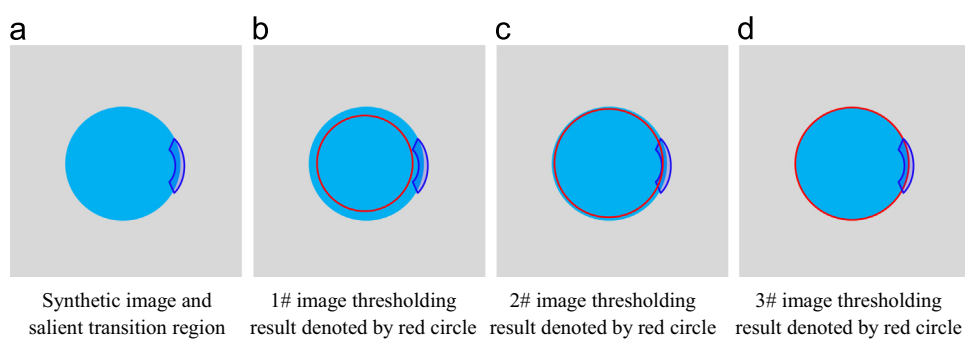

Fig. 5. Image thresholding results of a synthetic image obtained by three thresholds.

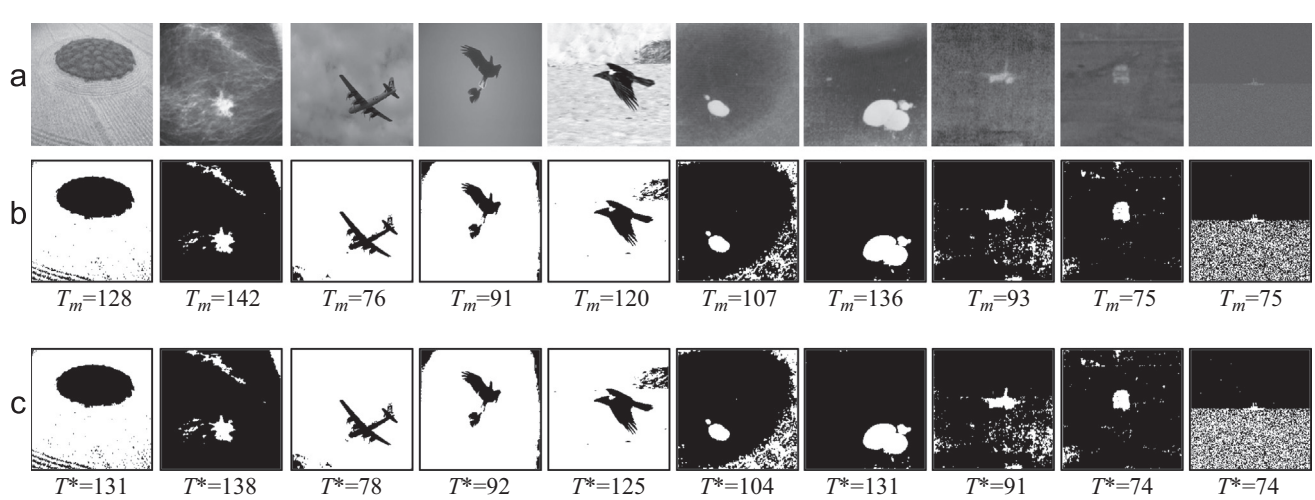

Fig. 6. Image thresholding results obtained by the gray level mean (T_m) of pixels in transition regions and the determined threshold (T^*) by the proposed method.

(1) Obtain image thresholding result B_{T^*} via the following formula:

$$B_{T^*}(i,j) = \begin{cases} 1 & \text{if } f(i,j) > T^*, \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

(2) Perform a reversion operation on B_{T^*} according to the similar criterion defined in Eq. (16), where B_t is replaced with B_{T^*}. Here, the reversion operation aims to guarantee that 1 and 0 denote object pixel and background pixel in B_{T^*}, respectively.

(3) Label each connected object region with a unique number in B_{T^*} by using Matlab function "bwlabel".
(4) Calculate the number of intersected pixels between each connected object region and salient transition region. If the number is equal to 0, the current object region is removed, i.e., setting gray levels of all pixels in current object region of B_{T^*} to zeros.
(5) Take the refined B_{T^*} as final segmentation result of the proposed method.

4. Experimental results

To evaluate the performance of the proposed method, 170 single-object real world images were chosen as testing samples. These samples are divided into two groups. The first group is composed of 10 typical images with resolutions of 200×200. Original images and their ground truths (manual segmentation results) are shown in Fig. 2(a) and (b), respectively. The second group consists of 160 images with different resolutions. Original images and their ground truths are shown in Figs. 7 and 8, respectively. The proposed method was compared with several types of image segmentation techniques, i.e., edge detector (Canny [8]), graph cuts (Ncut [11] and Isocut [12]), global thresholding (PWT [23]), local thresholding (Niblack [25]), hybrid image binarization (DTIB [31]), and transition region-based image thresholding (LE [33], LGLD [34] and MLE [35]).

Image segmentation results are quantitatively evaluated by a common measure, i.e., misclassification error (ME) [37]. ME regards image segmentation as a pixel classification process. It measures the percentage of background pixels that are incorrectly classified into foreground (object) pixels, and conversely, object pixels erroneously classified into background pixels. For a two-class segmentation issue, ME is defined as

$$\mathrm{ME} = 1 - \frac{|B_O \cap B_T| + |F_O \cap F_T|}{|B_O| + |F_O|}, \tag{23}$$

where B_O and F_O are the background and foreground of the ground truth (manual segmentation result), B_T and F_T are the background and foreground of an image segmentation result respectively, and $|\cdot|$ cardinality of a set. The value of ME varies between 0 for a perfectly classified image to 1 for a totally erroneously classified one. A lower value means better segmentation accuracy.

Among the reference methods, segmentation results of Canny, Ncut and Isocut are edge images as exhibited in Fig. 1(b)–(d). These edge images cannot be measured by ME. To perform quantitative performance comparison on Canny, Ncut and Isocut, we converted their segmentation results represented by edge images to binary region segmentation results similar with image thresholding results. Specifically, an edge image is first filled to generate segmentation regions by Matlab function "imfill". Then, image pixels corresponding to edge pixels in the edge image are converted into background pixels for avoiding object regions far away from each other adhered together by the edge pixels. Finally, only the largest object region is kept as final region segmentation result, because our research focuses on single-object segmentation. Taking the first image in Fig. 1(a) as a sample, Fig. 9 shows the whole process of converting segmentation result of Isocut to region segmentation result. Fig. 10 exhibits converted region segmentation results of Canny, Ncut and Isocut on the 10 images in the first group. Note that, during performing quantitative analysis on region segmentation results of the above three methods, we ensure that the object and the background in each region segmentation result have the same binary values as compared to the manual segmentation result.

In our experiments, Canny [8] is performed by Matlab function "edge". Stop criteria of Ncut [11] and Isocut [12] were set to 2×10^{-2} and 1×10^{-5}, respectively. Parameters of PWT [23], Niblack [25] and DTIB [31] refer to their own literatures. Differing from general image thresholding, transition region (TR)-based methods need extract image transition regions, which involves two parameters, i.e., neighborhood size (*Nsize*) and α. For one of LGLD [33], LE [34] and MLE [35], the best combination of both parameters is first found by minimizing mean misclassification error (MME) value on the 160 images in the second group, and the best parameter combination was adopted in all experiments. The proposed method also adopted its best parameter combination of $\{Nsize = 7 \times 7, w = 0, \alpha = 0.05, \beta = 0.1\}$ in all experiments. Note that, after constructing transition region description matrices, each TR-based method extracts transition region in the same way as the proposed method. This ensures that the number of transitional pixels obtained by each TR-based method is equal, which provides convenience and fairness for performance comparison between TR-based methods. All experiments are performed on a notebook PC with 1.7G Intel Core i5-3317U CPU and 4G RAM.

4.1. Parameter selection and other discussion

The proposed method has four parameters, i.e., neighborhood size (*Nsize*), w, α and β.The parameter w balances contributions of normalized local complexity and normalized local variance in transition region description. When $w=1$, the descriptor *NLCV* degenerates to local complexity. On the contrary, when $w=0$, *NLCV* degenerates to local variance. The parameter α denotes the ratio of transitional pixels over all image pixels. The parameter β is used to determine the candidate threshold interval R when searching for the segmentation threshold T^*. In our experiments, *Nsize* is selected from $\{3 \times 3,\ 5 \times 5,\ 7 \times 7,\ 9 \times 9,\ 11 \times 11,\ 13 \times 13,\ 15 \times 15\}$. The parameter w is selected from {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1}. The parameter α is selected from {0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1}. The parameter β is selected from {0.1, 0.2, 0.3, 0.4, 0.5}.

To study the impact of using transition regions or salient transition region to calculate T_{m} defined in Eq. (11) on segmentation accuracy, two versions of the proposed method are developed. One version, termed as RIB_T, uses gray level mean of all pixels in transition regions as T_m when calculating the gray level interval R in Section 3.3. The other version, termed as RIB_S, uses gray level mean of all pixels in salient transition region as T_m when calculating the gray level interval R. In addition, to study the impact of using local complexity or not on segmentation accuracy, we respectively recorded mean misclassification error (MME) values of both proposed methods under all parameter combinations considering local complexity ($w \neq 0$) or not ($w=0$). We conducted both methods on the 160 images in the second group.

To save space, only MME values of both methods under two key parameter combinations are listed in Table 1. Table 1 shows that the MME (0.077) corresponding to the best parameter combination of RIB_T using local complexity ($w \neq 0$) is higher than the MME (0.067) to the best parameter combination of RIB_T not using local complexity ($w=0$). Conversely, the MME (0.066)

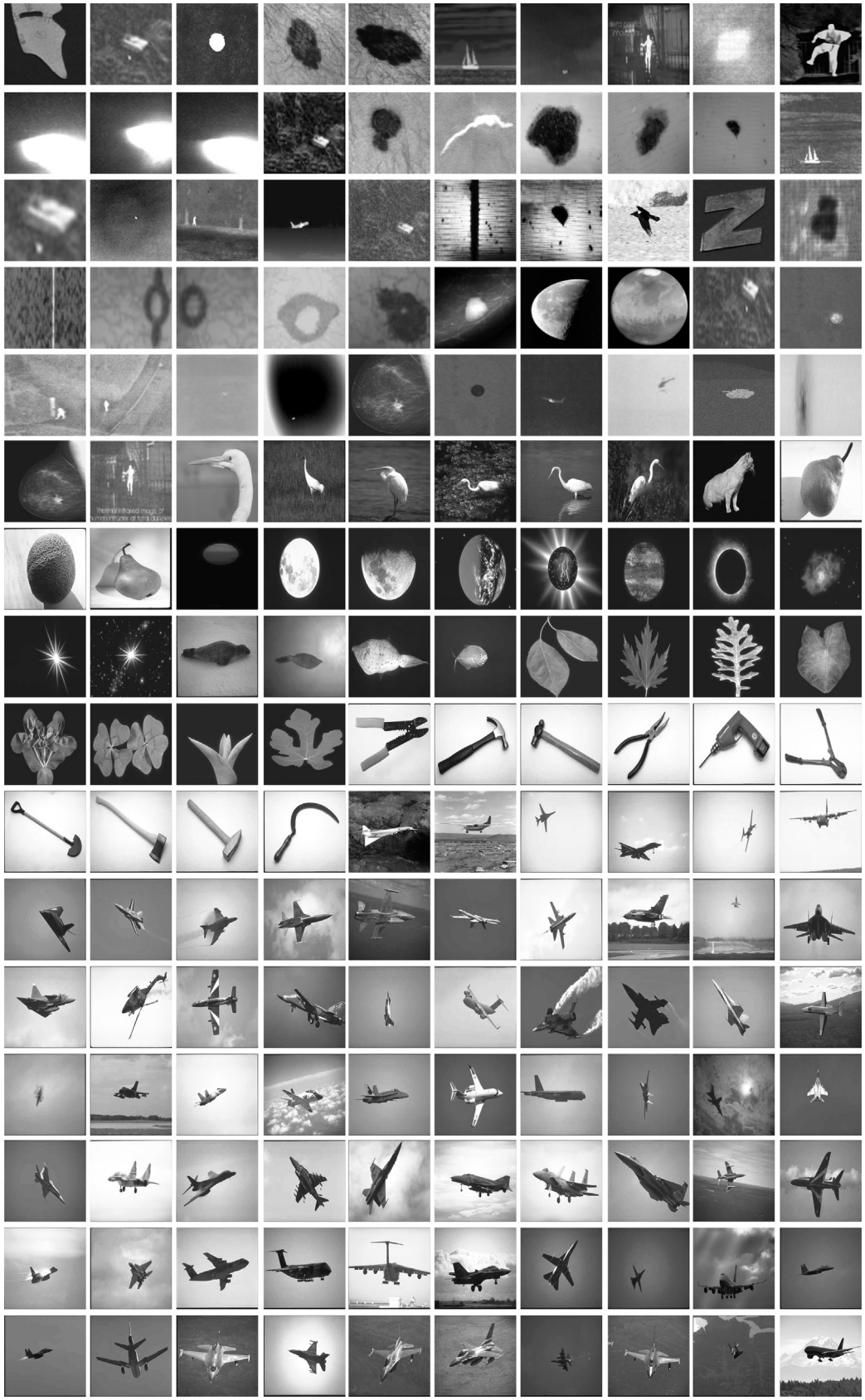

Fig. 7. The second group of original images.

corresponding to the best parameter combination of RIB_S using local complexity ($w \neq 0$) is lower than the MME (0.073) to the best parameter combination of RIB_ S not using local complexity ($w=0$). Thus, the use of local complexity decreases segmentation accuracy of RIB_T, but improves segmentation accuracy of RIB_S. The underlying reason is that the use of local complexity improves

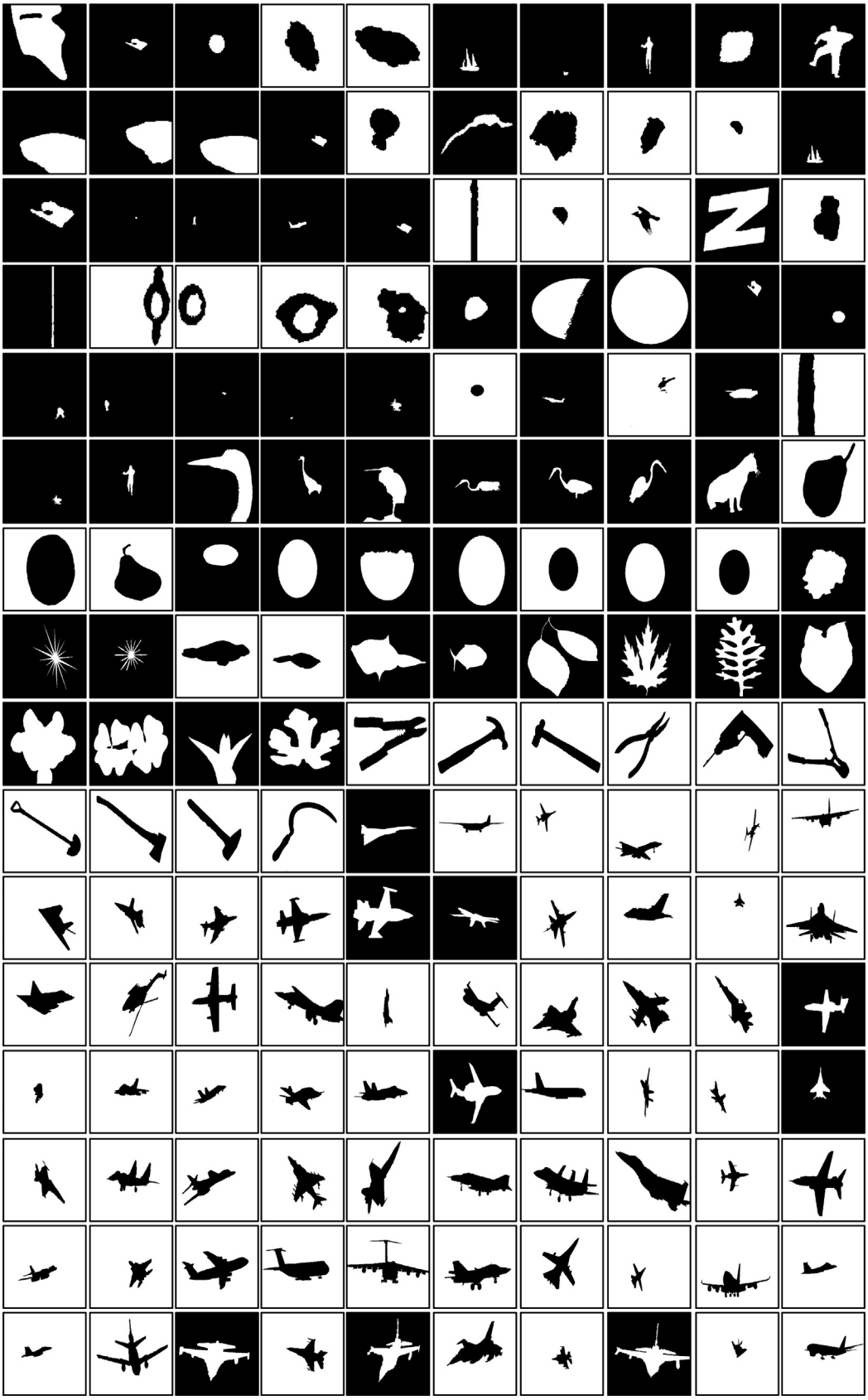

Fig. 8. Manual segmentation results corresponding to the second group of original images.

Fig. 9. Flow chart of converting segmentation result (edge image) into region segmentation result.

salient transition region extraction, but brings a few fake transition regions. These fake transition regions decrease the segmentation accuracy of RIB_T, and improved salient transition region increases the segmentation accuracy of RIB_S.

Under all parameter combinations including $w \neq 0$ and $w=0$, RIB_T obtains the smallest MME (0.067) at the parameter combination of $\{Nsize=7\times7,\ w=0,\ \alpha=0.05,\ \beta=0.1\}$, and RIB_S obtains the smallest MME (0.066) at the parameter combination of $\{Nsize=15\times15,\ w=0.4,\ \alpha=0.08,\ \beta=0.1\}$. RIB_S has slight improvement over RIB_T on segmentation accuracy, because RIB_S decreases MME by 0.001. But, the slight improvement occurs larger neighborhood size ($Nsize=15\times15$) and additional calculation of local complexity matrix in comparison with RIB_T. To reduce computational complexity and algorithm parameters, RIB_T with the parameter combination of $\{Nsize=7\times7,\ w=0,\ \alpha=0.05,\ \beta=0.1\}$ is used in our experiments, and is termed as RIB. Note that, RIB does not consider local complexity in transition region extraction, and uses gray level mean of all pixels in transition regions as T_m when searching for the segmentation threshold T^*.

After choosing the best parameter combination of $\{Nsize=7\times7,\ w=0,\ \alpha=0.05,\ \beta=0.1\}$ based on the 160 images in the second group, we adopted a leave-one-out manner to study the influence of each parameter on segmentation accuracy and to give the guidance on their appropriate ranges. Fig. 11 shows mean misclassification error (MME) values obtained by applying the proposed method (RIB_T) with three fixed parameters and one variable parameter to the 160 images in the second group. Fig. 11 (a) shows that the lowest MME value (0.067) was obtained with a neighborhood size (*Nsize*) of 7×7. Neighborhood size needs to be large enough for description of image transition region, but should not be too large to miss the localization. Fig. 11(b) and (d) shows that the lowest MME values (0.067) were obtained with $w=0$ and $\beta=0.1$, respectively. The increase of w and β decreases segmentation accuracy. Fig. 11(c) shows that the lowest MME value (0.067) was obtained with $\alpha=0.05$. In summary, the parameter α needs to be large enough for extracting enough transition region, but should not be too large to generate too many fake transition regions.

4.2. The first group of experiments

In this group of experiments, 10 typical single-object images were selected as testing samples. Quantitative comparisons on ME values obtained by various methods are listed in Table 2. From the table, it can be seen that RIB obtains the highest segmentation accuracy with the lowest ME values on the first 8 images. For the image *Car*, RIB obtains the second lowest ME value. For the image *Ship*, RIB obtains the sixth lowest ME value. In summary, RIB achieves the best segmentation accuracy in most cases. Performance judgment with various methods can also be evidenced by visual segmentation results shown in Fig. 1 and Fig. 2. Fig. 2 shows that RIB obtains satisfactory segmentation results on the first 9 images. Conversely, RIB misclassifies a lot of background pixels in the last image into object pixels. The failure is due to that the noise region composed of misclassified background pixels cannot be removed by RIB in its refining process of image thresholding result, because the noise region adheres to the true object region.

Table 1
Mean misclassification error (MME) values of the proposed methods under two key parameter combinations.

Key parameter combinations	*Nsize*	*w*	*α*	*β*	MME
The best parameter combination of RIB_T using local complexity ($w\neq0$)	13×13	0.1	0.08	0.1	0.077
The best parameter combination of RIB_T not using local complexity ($w=0$)	7×7	0	0.05	0.1	0.067
The best parameter combination of RIB_S using local complexity ($w\neq0$)	15×15	0.4	0.08	0.1	0.066
The best parameter combination of RIB_S not using local complexity ($w=0$)	13×13	0	0.06	0.1	0.073

Fig. 1 shows that Canny [8] captures too many undesirable object and background details, which causes bad visual effect and even non-identification of object. Ncut [11] can only get good segmentation results on simple images with high contrast, such as Crow and Airhole2. Isocut [12] obtains good segmentation results on five images. However, Isocut fails to achieve satisfactory results on those images with complex background or low contrast. Fig. 2 shows that DTIB [31], Niblack [25] and MLE [35] obtain bad segmentation results. The failure of DTIB is due to its great dependency on the edge detection result of Canny. The failure of Niblack is attributed to the difficulty of parameters' adaptive selection in the process of performing local thresholding. The failure of MLE mainly comes from its terrible transition regions caused by its inappropriate image transformation. LGLD [34] and LE [33] also generate unsatisfactory results in most cases. This is simultaneously caused by both inaccurate transition regions and the limitations inherited from global image thresholding.

4.3. The second group of experiments

The second group of experiments was conducted on 160 single-object real world images with different resolutions, and these images are named from 1 to 160. Original images and corresponding manual ideal segmentation results (ground truths) are exhibited in Figs. 7 and 8, respectively. To save space, visual segmentation results obtained by various methods are not listed in this paper.

Quantitative performance comparisons of various methods are listed in Table 3 and Table 4. To save space, Table 3 only lists mean misclassification error (MME) value and standard deviation of ME values obtained by applying each method to the 160 images. ME values obtained by applying each method to every image are provided as an attachment called "MEs.doc". From Table 3, it can be seen that the MME value of RIB is dramatically lower than those of other methods. This means that RIB has the best average segmentation accuracy. In addition, RIB obtains the fourth lowest standard deviation of ME values. This indicates that RIB has better

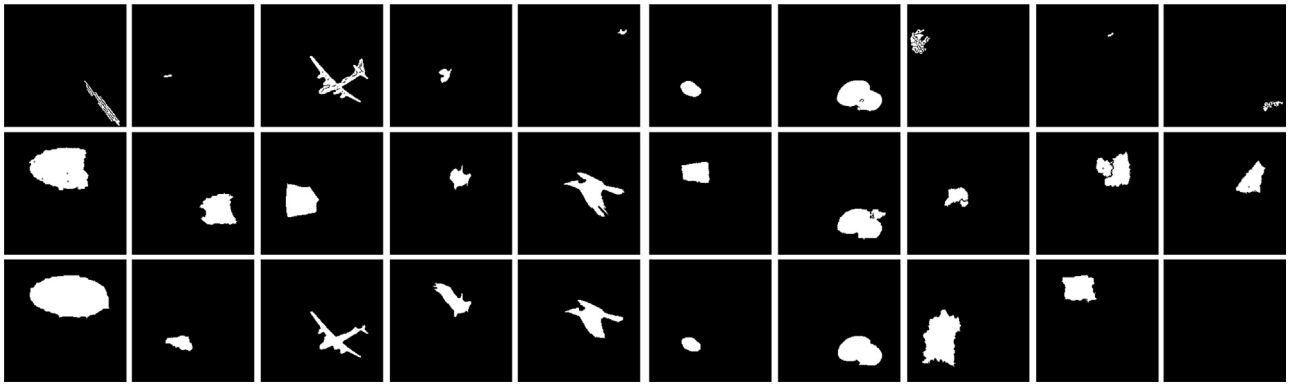

Fig. 10. Region segmentation results of Canny [8], Ncut [11] and Isocut [12].

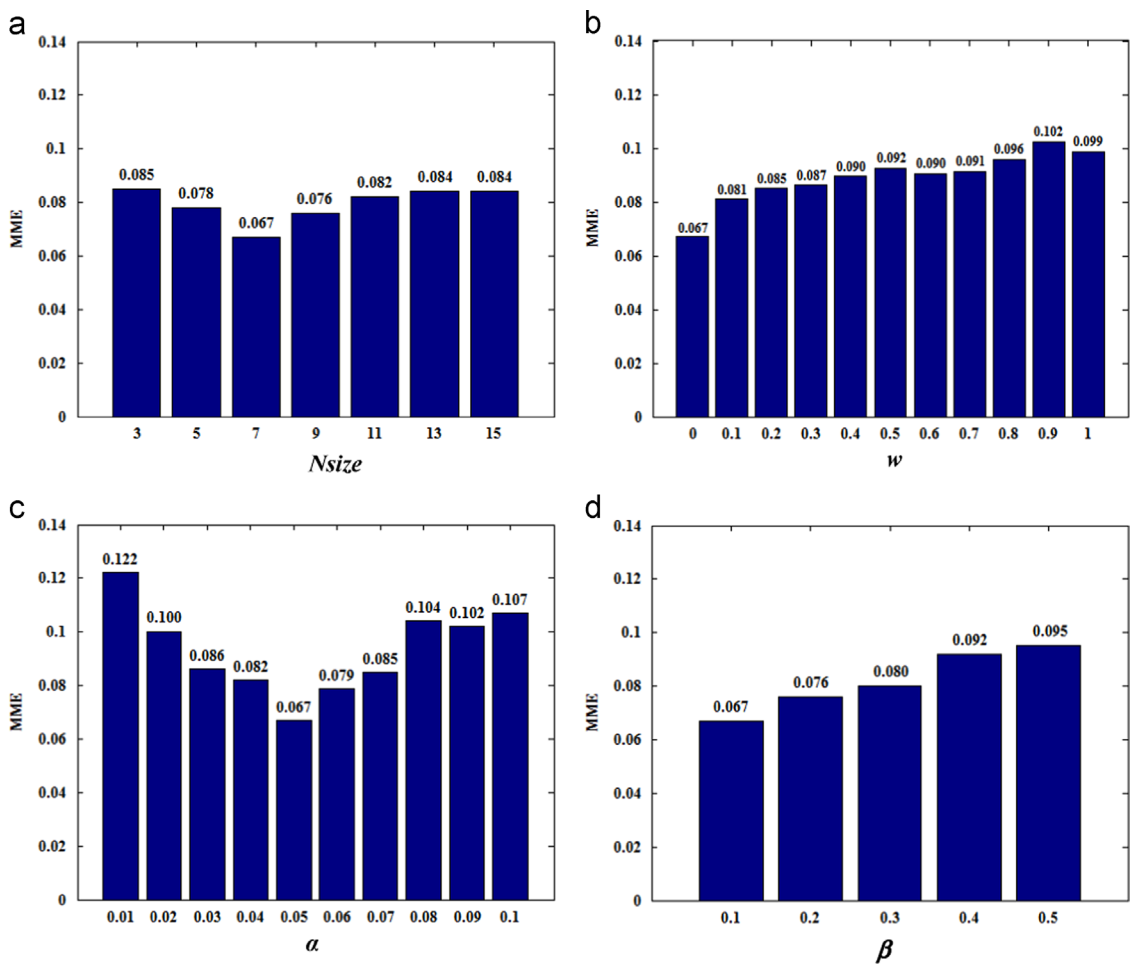

Fig. 11. Mean misclassification error (MME) values obtained by applying RIB_T with three fixed parameters and one variable parameter under the best parameter combination of $\{Nsize=7\times 7,\ w=0,\ \alpha=0.05,\ \beta=0.1\}$ to the 160 images in the second group.

stability than its counterparts excluding Canny, Niblack and Ncut. Although Canny, Niblack and Ncut have better stability with lower standard deviations, they have worse segmentation accuracy with higher MME values in comparison with RIB.

To compare the performance of various methods on the 160 images in detail, we sorted ME values obtained by applying each method to every image, and performed statistical analysis on the ranking of each method. Statistical results are listed in Table 4, where each row indicates the ratios of various methods of obtaining corresponding ranking ME values among the 160 images. From the table, it can be seen that RIB obviously outperforms other methods. Among all 10 methods, RIB obtains the lowest ME values on 51.875% testing samples, and achieves the top 3 ranking on 80% samples. In short, Table 4 demonstrates the effectiveness and robustness of RIB again.

4.4. Computational time

The computational time of RIB is related with several factors such as image size and neighborhood size. Computational complexity of RIB is first theoretically analyzed in this section. To segment an image of n pixels with a 7×7 local neighborhood, RIB

Table 2
Quantitative comparison on ME (Misclassification Error) for the first group of images.

Images	Canny	Ncut	Isocut	PWT	DTIB	Niblack	LGLD	LE	MLE	RIB
Field	0.188	0.048	0.010	0.053	0.209	0.392	0.014	0.009	0.168	0.003
Mammography	0.019	0.072	0.031	0.036	0.222	0.518	0.037	0.028	0.393	0.006
Airplane1	0.016	0.114	0.017	0.017	0.089	0.316	0.011	0.011	0.155	0.006
Eagle	0.043	0.033	0.017	0.009	0.874	0.276	0.008	0.008	0.007	0.003
Crow	0.067	0.010	0.015	0.026	0.185	0.326	0.016	0.024	0.062	0.001
Airhole1	0.003	0.051	0.005	0.014	0.273	0.561	0.005	0.061	0.225	0.002
Airhole2	0.013	0.009	0.015	0.011	0.235	0.520	0.004	0.005	0.279	0.002
Airplane2	0.032	0.038	0.112	0.957	0.523	0.574	0.177	0.017	0.546	0.004
Car	0.012	0.062	0.058	0.001	0.430	0.564	0.039	0.019	0.430	0.009
Ship	0.008	0.038	0.004	0.019	0.339	0.577	0.192	0.286	0.372	0.282

Table 3
Quantitative comparison on MME (Mean Misclassification Error) and standard deviation of ME (Misclassification Error) values for the second group of images.

	Canny	Ncut	Isocut	PWT	DTIB	Niblack	LGLD	LE	MLE	RIB
MME	0.107	0.167	0.117	0.156	0.234	0.405	0.107	0.127	0.194	0.067
Standard deviation	0.100	0.132	0.189	0.254	0.216	0.103	0.164	0.180	0.176	0.151

Table 4
Statistical results (%) of ranking on ME (Misclassification Error) values obtained by applying various methods to the second group of images.

Ranking of ME	Canny	Ncut	Isocut	PWT	DTIB	Niblack	LGLD	LE	MLE	RIB
1	6.875	3.750	9.375	3.125	1.875	0.000	8.125	2.500	12.500	51.875
2	8.750	3.750	25.625	9.375	6.250	0.000	12.500	16.250	5.000	12.500
3	13.750	8.750	8.125	10.625	2.500	0.000	21.250	15.625	3.750	15.625
4	5.625	8.125	10.625	18.750	5.000	3.125	16.875	18.750	6.250	6.875
5	6.875	7.500	9.375	20.000	10.625	1.875	13.750	17.500	8.750	3.750
6	10.625	7.500	8.750	15.625	20.000	1.875	12.500	10.625	10.000	2.500
7	18.125	17.500	11.250	6.250	10.000	3.750	11.875	6.875	11.875	2.500
8	20.000	13.125	5.000	3.125	21.250	6.875	2.500	8.125	18.125	1.875
9	5.625	18.125	5.625	3.750	14.375	28.750	0.000	1.250	21.875	0.625
10	3.750	11.875	6.250	9.375	8.125	53.750	0.625	2.500	1.875	1.875

Table 5
Average running time (in seconds) on the first group of images.

Canny	Ncut	Isocut	PWT	DTIB	Niblack	LGLD	LE	MLE	RIB	FRIB
0.129	7.140	0.929	45.206	4.095	2.979	0.999	9.945	4.847	1.375	0.175

first calculates all pixels' local variances and sorts them for transition region extraction, whose complexities are $O(49n)$ and $O(n\log n)$, where $\log n \ll n$. Then, RIB extracts salient transition region from transition regions, which has complexity $O(n_t)$, where n_t is the number of transitional pixels and $n_t \ll n$. Next, RIB requires $O(kn)$ to search for a threshold from k candidate thresholds for image thresholding, where $k \ll 256 \ll n$. Finally, RIB refines image thresholding result by removing fake object regions, which has complexity $O(n)$. Thus, the total complexity of RIB is about $O((51+k+\log n)\cdot n)$.

To compare computational efficiency of various methods, we specifically recorded their average computational time on the 10 images with resolutions of 200×200 in the first group. Experimental results are listed in Table 5. In addition, the table compares RIB and its fast version (FRIB). Motivated by the method in [38], to save running time, FRIB pre-calculates variance of each patch as a map of local variances. Table 5 shows that RIB is slower than Canny, FRIB, Isocut and LGLD, but is faster than the other six methods. Furthermore, FRIB is only slower than Canny, and it is 10 times faster than RIB. Overall, the proposed method achieves satisfactory image segmentation results with very low computational cost.

5. Conclusions and future works

In this paper, we proposed a novel hybrid single-object image segmentation method based on salient transition region. The proposed method exploits transition regions from a unique perspective in comparison with conventional transition region-based image thresholding. Specifically, the proposed method presents a new concept (i.e., salient transition region), and designs a simple and effective criterion to measure saliency of transition regions. The proposed method also presents a scheme of using both transition regions and salient transition region to find a more appropriate segmentation threshold for accurately segmenting the object. Furthermore, the proposed method presents a scheme of refining image thresholding result by removing fake object regions that non-overlap with salient transition region. Experimental results on a series of single-object real world images demonstrate that the proposed method significantly improves segmentation

accuracy and robustness in comparison with existing transition region-based image thresholding methods, as well as other segmentation techniques.

However, the proposed method does not achieve high accuracy in segmenting images whose image thresholding results contain noise region adhered to the true object. In the future, we will try to resolve this issue via utilizing image transition regions more adequately.

Conflict of interest

None declared.

Acknowledgment

This work is supported by National Natural Science Foundation of China (Nos. 61202318, 61463008, 61202313, 61203244 and 61370163), Program for New Century Excellent Talents in Fujian Province University (NCETFJ), Technology Project of Provincial University of Fujian Province (No. JK2014040), Fujian College Outstanding Young Research Personnel Training Project (No. JA13247) and Natural Science Foundation of Jiangsu Province (No. BK20131342).

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.patcog.2015.10.009.

References

[1] L. Dai, J. Ding, J. Yang, Inhomogeneity-embedded active contour for natural image segmentation, Pattern Recognit. 48 (8) (2015) 2513–2529.
[2] Y. Zhang, M.A. Mottaleb, Z. He, Unsupervised segmentation of highly dynamic scenes through global optimization of multiscale cues, Pattern Recognit. 48 (11) (2015) 3477–3487.
[3] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1–8.
[4] T.H. Min, R.H. Park, Eyelid and eyelash detection method in the normalized iris image using the parabolic Hough model and Otsu's thresholding method, Pattern Recognit. Lett. 30 (12) (2009) 1138–1143.
[5] Y.T. Pai, Y.F. Chang, S.J. Ruan, Adaptive thresholding algorithm: efficient computation technique based on intelligent block detection for degraded document images, Pattern Recognit. 43 (9) (2010) 3177–3187.
[6] Z. Li, C. Liu, G. Liu, X. Yang, Y. Cheng, Statistical thresholding method for infrared images, Pattern Anal. Appl. 14 (2) (2011) 109–126.
[7] O.V. Senyukova, Segmentation of blurred objects by classification of isolabel contours, Pattern Recognit. 47 (12) (2014) 3881–3889.
[8] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell. 8 (6) (1986) 679–698.
[9] J.M.S. Prewitt, Object Enhancement and Extraction, Picture Processing and Psychopictorics, Academic Press, Waltham, Massachusetts, 1970.
[10] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison-Wesley Publishing Company, Reading, Massachusetts, 1992.
[11] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[12] L. Grady, E.L. Schwartz, Isoperimetric graph partitioning for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 28 (3) (2006) 469–475.
[13] B. Peng, L. Zhang, D. Zhang, A survey of graph theoretical approaches to image segmentation, Pattern Recognit. 46 (3) (2013) 1020–1038.
[14] S. Yin, X. Zhao, W. Wang, M. Gong, Efficient multilevel image segmentation through fuzzy entropy maximization and graph cut optimization, Pattern Recognit. 47 (9) (2014) 2894–2907.
[15] F. Galasso, M. Keuper, T. Brox, B. Schiele, Spectral graph reduction for efficient image and streaming video segmentation, in: Computer Vision and Pattern Recognition (CVPR), 2014, pp. 4321–4328.
[16] Z. Li, J. Yang, G. Liu, Y. Cheng, C. Liu, Unsupervised range-constrained thresholding, Pattern Recognit. Lett. 32 (2) (2011) 392–402.
[17] M. Gong, Y. Liang, J. Shi, W. Ma, J. Ma, Fuzzy C-means clustering with local information and keneral metric for image segmentation, IEEE Trans. Image Process. 22 (2) (2013) 573–584.
[18] P. Yu, A.K. Qin, D.A. Clausi, Unsupervised polarimetric SAR image segmentation and classification using region growing with edge penalty, IEEE Trans. Geosci. Remote Sens. 50 (4) (2012) 1302–1317.
[19] N. Thakoor, G. Jean, V. Devarajan, Multistage branch-and-bound merging for planar surface segmentation in disparity space, IEEE Trans. Image Process. 17 (11) (2008) 2217–2226.
[20] S.N. Sulaiman, N.A.M. Isa, Adaptive fuzzy-K-means clustering algorithm for image segmentation, IEEE Trans. Consum. Electron. 56 (4) (2010) 2661–2668.
[21] Z. Hou, Q. Hu, W.L. Nowinski, On minimum variance thresholding, Pattern Recognit. Lett. 27 (14) (2006) 1732–1743.
[22] M.P. Albuquerque, I.A. Esquef, A.R.G. Mello, Image thresholding using Tsallis entropy, Pattern Recognit. Lett. 25 (9) (2004) 1059–1065.
[23] S. Wang, F. Chung, F. Xiong, A novel image thresholding method based on parzen window estimate, Pattern Recognit. 41 (1) (2008) 117–129.
[24] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, J. Electron. Imaging 13 (1) (2004) 146–165.
[25] W. Niblack, An introduction to digital image processing, Prentice-Hall, Englewood Cliffs, N.J (1986), p. 115–116.
[26] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, Pattern Recognit. 33 (2) (2000) 225–236.
[27] W. Zou, Z. Liu, K. Kpalma, J. Ronsin, Y. Zhao, N. Komodakis, Unsupervised joint salient region detection and object segmentation, IEEE Trans. Image Process. 24 (11) (2015) 3858–3873.
[28] Y. Cheng, X. Hu, J. Wang, Y. Wang, S. Tamura, Accurate vessel segmentation with constrained B-snake, IEEE Trans. Image Process. 24 (8) (2015) 2440–2455.
[29] H. Wang, J.W. Suh, S.R. Das, J.B. Pluta, C. Craige, P.A. Yushkevich, Multi-atlas segmentation with joint label fusion, IEEE Trans. Pattern Anal. Mach. Intell. 35 (3) (2013) 611–623.
[30] L. Wang, G. Hua, J. Xue, Z. Gao, N. Zheng, Joint segmentation and recognition of categorized objects from noisy web image collection, IEEE Trans. Image Process. 23 (9) (2014) 4070–4086.
[31] Q. Chen, Q. Sun, P.A. Heng, D. Xia, A double-threshold image binarization method based on edge detector, Pattern Recognit. 41 (4) (2008) 1254–1267.
[32] Y.J. Zhang, J.J. Gerbrands, Transition region determination based thresholding, Pattern Recognit. Lett. 12 (1) (1991) 13–23.
[33] C. Yan, N. Sang, T. Zhang, Local entropy-based transition region extraction and thresholding, Pattern Recognit. Lett. 24 (16) (2003) 2935–2941.
[34] Z. Li, C. Liu, Gray level difference-based transition region extraction and thresholding, Comput. Electr. Eng. 35 (5) (2009) 696–704.
[35] Z. Li, D. Zhang, Y. Xu, C. Liu, Modified local entropy-based transition region extraction and thresholding, Appl. Soft Comput. 11 (8) (2011) 5630–5638.
[36] C. Yan, N. Sang, T. Zhang, K. Zeng, Image transition region extraction and segmentation based on local complexity, J. Infrared Millim. Waves 24 (4) (2005) 312–316.
[37] W.A. Yasnoff, J.K. Mui, J.W. Bacus, Error measures for scene segmentation, Pattern Recognit. 9 (4) (1977) 217–231.
[38] P. Coupé, J.V. Manjón, V. Fonov, J. Pruessner, M. Robles, D.L. Collins, Patch-based segmentation using expert priors: application to hippocampus and ventricle segmentation, Neuroimage 54 (2) (2011) 940–954.

Zuoyong Li received the B.S. and M.S. degrees in computer science and technology from Fuzhou University, Fuzhou, China, in 2002 and 2006, respectively. He received the Ph.D. degree from the School of Computer Science and Technology at Nanjing University of Science and Technology, (NUST), Nanjing, China, in 2010. He is currently an associate professor in the Department of Computer Science of Minjiang University, Fuzhou, China. He has published several papers in international/national journals. His current research interest is image processing.

Guanghai Liu is currently a Professor in the College of Computer Science and Information Technology, Guangxi Normal University in China. He received Ph.D. degree from the School of Computer Science and Technology, Nanjing University of Science and Technology (NUST). His current research interests are in the areas of image processing, pattern recognition and artificial intelligence.

David Zhang received the B.Sc. degree in computer science from Peking University, Beijing, China. He received the M.Sc. degree in computer science, and the Ph.D. degree from Harbin Institute of Technology (HIT), Shenzhen, China, in 1982 and 1985, respectively. He received the second Ph.D. degree in electrical and computer engineering from

the University of Waterloo, Ontario, Canada, in 1994. He was a postdoctoral Fellow at Tsinghua University, Beijing, from 1986 to 1988, and then an associate professor at the Academia Sinica, Beijing. Currently, he is the head of the Department of Computing, and a chair professor at the Hong Kong Polytechnic University, Kowloon, Hong Kong, where he has been the founding director of the biometrics technology center supported by the government of the Hong Kong special administrative region since 1998. He also serves as a visiting chair professor in Tsinghua University, and adjunct professor in Peking University, Shanghai Jiao Tong University, Shanghai, HIT, and the University of Waterloo. He is the author of more than 10 books and 200 journal papers. He is the founder and editor-in-chief of the International Journal of Image and Graphics, and the editor of the book International Series on Biometrics (Springer). He is the editor of more than 10 international journals including IEEE transactions and Pattern Recognition. He was the organizer of the International Conference on Biometrics Authentication. Professor Zhang is a Croucher senior research fellow, distinguished speaker of the IEEE Computer Society, and fellow of both IEEE and International Association for Pattern Recognition (IAPR).

Yong Xu received his B.S. and M.S. degrees at Air Force Institute of Meteorology (China) in 1994 and 1997, respectively. He then received his Ph.D. degree in pattern recognition and intelligence system at the Nanjing University of Science and Technology (NUST) in 2005. From May 2005 to April 2007, he worked at Shenzhen graduate school, Harbin Institute of Technology (HIT) as a postdoctoral research fellow. Now he is a professor at Shenzhen graduate school, HIT. He also acts as a research assistant researcher at the Hong Kong Polytechnic University from August 2007 to June 2008. His current interests include pattern recognition, biometrics, and machine learning. He has published more than 40 scientific papers.