ELSEVIER

Pattern Recognition Letters 18 (1997) 963–974

Pattern Recognition Letters

Evaluation and comparison of different segmentation algorithms

Y.J. Zhang [1]

Department of Electronic Engineering Tsinghua University, 100084 Beijing, China

Received 21 January 1997; revised 17 June 1997; accepted 18 August 1997

Abstract

This paper presents an objective and quantitative study of segmentation algorithms. This study is distinguished from other studies by considering both evaluation and comparison, treating algorithms selected from distinct technique groups as well as using carefully designed synthetic images for the test experiments. All these characteristics make this study a general and effective one for revealing the performance of segmentation algorithms. © 1997 Elsevier Science B.V.

Keywords: Image segmentation; Segmentation evaluation; Segmentation comparison; Algorithm classification; Synthetic image; Performance assessment

1. Introduction

Image segmentation is an important process of image analysis. It consists of subdividing an image into its constituent parts and extracting these parts of interest (objects). A large variety of different segmentation algorithms have been developed (Pal and Pal, 1993). The evaluation and comparison of these algorithms turn to be important and even indispensable for rightly using these algorithms. Recently, several comparative studies of segmentation algorithms have been reported (Sahoo et al., 1988; Lee et al., 1990; Pal and Bhandari, 1993; Pal and Pal, 1993). The following comments about these works can be made:

(1) In all these studies, only the performance comparison among various algorithms is discussed and no intra-evaluation of particular algorithms has been reported. Though segmentation evaluation and segmentation comparison are closely related, they are in fact distinct matters. Segmentation evaluation is an intra-technique process. The purpose of evaluation for a specific algorithm is to quantitatively recognize its behavior in treating various images and/or to help appropriately setting its parameters regarding different applications to achieve the best performance of this algorithm. Segmentation comparison is an inter-technique process. The purpose of comparison for different algorithms is to rank their performance and to provide guidelines in choosing suitable algorithms according to applications as well as to promote new developments by effectively taking the strong points of several algorithms.

(2) In these studies, most segmentation algorithms treated are based on threshold selection. Two exceptions (two iterative pixel classification algorithms) were reported (Pal and Pal, 1993). However, the comparison made on these two algorithms is separated from the comparison made on the thresholding algorithms. Though the threshold selection based

[1] Email: zhangyj@img.ee.tsinghua.edu.cn.

0167-8655/97/$17.00
PII S0167-8655(97)00083-4

techniques are often used, the algorithms based on other principles are more and more popular (Pal and Pal, 1993). A complete evaluation and comparison of segmentation algorithms should take into account different types of techniques together.

(3) In most of these studies only particular real images were used for test purpose. Though real images may have some realistic characteristics, their "random" nature makes the studies less general and not suitable for different applications. Moreover, numerous indeterminate characteristics of real images make an analytic evaluation and an accurate comparison complicated. As a rule, the segmentation results can only be judged either by using manually segmented images as reference (Lee et al., 1990; Pal and Bhandari, 1993), or by visually comparing to the original images (Pal and Pal, 1993), or just applying quality measures corresponding to human intuition (Sahoo et al., 1988; Lee et al., 1990; Pal and Pal, 1993). This makes these studies more subjective than objective. Finally these studies can not be inter-compared as the images used are quite different.

This paper attempts to address these three points. It presents an evaluation and comparison study of different types of segmentation algorithms. This study is made with the help of carefully designed synthetic images and new performance criteria that are both objective and quantitative (Zhang and Gerbrands, 1994). In Section 2 the methods for segmentation evaluation and comparison are depicted. According to a classification scheme, several segmentation algorithms from different technique groups are selected for this study. Both evaluation and comparison of segmentation algorithms are then detailed. The evaluation studies and the result analyses are presented in Section 3. The comparison experiments and the result discussions are given in Section 4. Finally, the conclusions drawn from this study and advantages of our approach are summarized in Section 5.

2. Method description

2.1. Classification and selection of algorithms

A classification of algorithms into groups is a partition of a set into subsets. With reference to the definition of segmentation (Fu and Mei, 1981), we believe that an appropriate classification of segmentation algorithms should satisfy the following four conditions:

1. Every algorithm must be in a group.
2. All groups together can include all algorithms.
3. The algorithm in the same group should have some common properties.
4. The algorithm in different group should have certain distinguishable properties.

Classifications of algorithms are performed according to some classification criteria. The first two conditions imply that the classification criteria should be suitable for classifying all different algorithms. The last two conditions imply that the classification criteria should determine the representative properties of each algorithm group. Taking these conditions in mind, the following two criteria appear to be suitable for the classification of segmentation algorithms.

The gray level image segmentation is generally based on one of two basic properties of gray level values in images: discontinuity and similarity (Conzalez and Woods, 1992). So two categories of algorithms can be distinguished: the boundary based ones that detect object contours explicitly by using the discontinuity property and the region based ones that locate object areas explicitly according to the similarity property. These two categories can be considered as complementary. On the other hand, according to the processing strategy, segmentation algorithms can be divided into the sequential class and parallel class (Rosenfeld, 1981). In the former, some cues from the early processing steps are taken for the subsequent steps. While in the latter, all decisions are made independently and simultaneously. Both strategies are also complementary from the processing point of view.

Combining the two types of categorizations four groups of techniques, G1, G2, G3 and G4 can be defined as shown in Table 1. Such a classification satisfies the above four conditions. These four groups

Table 1
Classification of segmentation techniques

Segmentation techniques	Parallel strategy	Sequential strategy
Boundary based	Group 1 (G1)	Group 2 (G2)
Region based	Group 3 (G3)	Group 4 (G4)

can include all existing segmentation algorithms, for example, surveyed by Fu and Mei (1981) as well as by Pal and Pal (1993).

To make this study complete, one algorithm from each group has been selected as representative examples:

CEBC (G1): Canny operator Edge detecting and Boundary Closing (see below).

DPCS (G2): Dynamic Programming technique for Contour Searching (Gerbrands, 1988).

IHCA (G3): Improved Histogram Concavity Analysis (Zhang et al., 1990).

SMG (G4): Split, Merge and Group approach (Strasters and Gerbrands, 1991).

Moreover, another algorithm called T-B which stands for Thresholding with the Best results (see Section 4.1) is also tested along with the above four algorithms for the comparison purpose. This algorithm, as a thresholding technique, belongs to the group G3. The particular implementation of those five algorithms will be described in the following sections when they are treated.

2.2. Construction of test images

The design and generation of synthetic test images for different segmentation evaluation and comparison tasks have been detailed in reference (Zhang and Gerbrands, 1992a). Two image subsets used in this study, namely the subset Shape and the subset Size, are shown together in Fig. 1. These images are size of 256×256, with 256 gray levels. In Fig. 1, the left 8 columns belong to the subset Size and the right 4 columns belong to the subset Shape. Note that the most right column of the subset Size is also the most left column of the subset Shape. This makes all experiments made with these two subsets internally connected.

In the subset Size, the diameter of disk objects is varied from column to column. The eight objects, from left to right, have diameters 20, 28, 40, 50, 64, 90, 112 and 128, respectively. They are labeled from #1 to #8. In the subset Shape, the form of objects changes from the circle to elongated ellipse. The eccentricity (E) values of these four objects are 1.0, 1.5, 2.0 and 2.5, respectively. The eccentricity is obtained on the basis of fitting an ellipse to the object boundary:

$$E = Mal/Mil, \tag{1}$$

where Mal and Mil are the major and minor axis length of fitting ellipse, respectively. The diameter of the circular object is 128. Other objects, though with different shape, are adjusted to have the similar size as that of the circular one to avoid the influence of object size over segmentation results.

To make the images more realistic, one 3×3 average low-pass filter is applied for producing a transition region between objects and background (Zhang and Gerbrands, 1991). Then, zero-mean Gaussian noise is added to simulate noisy effect. The noise samples have been generated with four standard deviations 4, 8, 16 and 32, respectively. The

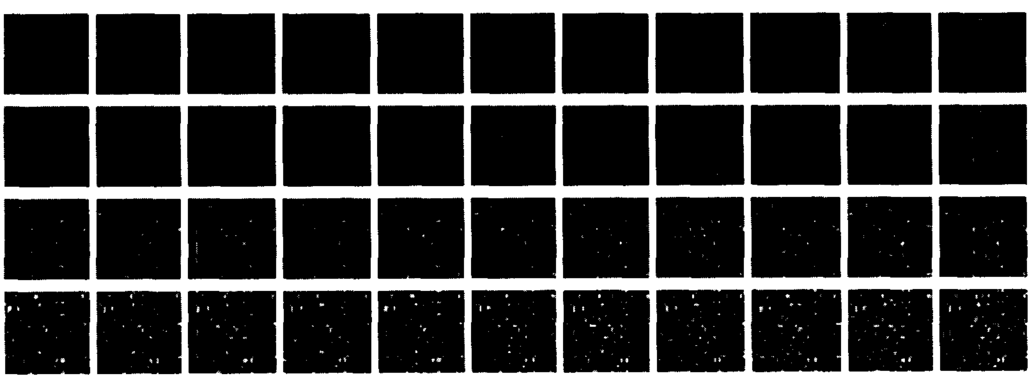


Fig. 1. Synthetic test images: subset Size (left 8 columns) and subset Shape (right 4 columns).

Signal-to-Noise Ratio (SNR) of images is defined as (Kitchen and Rosenfeld, 1981):

$$SNR = (h/\sigma)^2, \tag{2}$$

where h is the gray level difference between the object and background regions and σ is the standard deviation of the noise. Since the gray level of objects is 144 and the gray level of background is 112 in the noise-free images, the SNR levels of the four image rows in Fig. 1, from top to bottom, are 64, 16, 4, and 1, respectively. These values cover the range of many applications (Kitchen and Rosenfeld, 1981). To cope with the randomness of noise, ten noisy images are separately generated for each SNR level. Each image shown in Fig. 1 is one of these ten images.

2.3. Characterization of performance

To characterize the quality of segmented image and the performance of applied algorithms, certain judging criteria are needed. In this study the objective and quantitative criteria as discussed in Zhang and Gerbrands (1994) are used. Let R_f denote the feature value obtained from a reference image and S_f denote the feature value measured from the segmented image, the performance criteria, which are called Relative Ultimate Measurement Accuracy of an object feature ($RUMA_f$) can be computed by:

$$RUMA_f = (|R_f - S_f|/R_f) \times 100\%. \tag{3}$$

The values of $RUMA_f$ are inversely proportional to the segmentation results: the smaller the values, the better is the quality (regarding to the feature used). It has been shown that among all groups of existing criteria for segmentation evaluation, the group with RUMA is the best to precisely judge the quality of segmentation results (Zhang, 1996). In general the perfect segmentation should extract the object as it was generated and a better segmentation could provide a result closer to the perfect one.

Features used in Eq. (3) can be selected according to the segmentation goals as well as evaluation and comparison requirements. Various object features can be employed (two examples are shown in the next sections) so that different situations can be covered. In practical evaluation and comparison experiments, the algorithms selected are first applied to the test images, and then S_f can be measured from the segmented images. The R_f can be obtained by a similar process from the reference image produced in image generation process. With both S_f and R_f in hand, the $RUMA_f$ can be readily computed.

3. Segmentation evaluation

For one segmentation algorithm, two types of evaluation can often be distinguished. One is carried out with the same parameter setting of this algorithm for segmenting multiple images. That is, the ability and consistency of the algorithm in treating images with different contents and/or acquired under various conditions are evaluated. We will show such an example by evaluating the DPCS algorithm with the help of the image subset Shape and using the eccentricity as object feature in Section 3.1. Another one is carried out by giving different values to the algorithm's parameters for segmenting some comparable images and then evaluating the influence of multiple settings of the algorithm over its performance. That is, the adaptability and the best performance of this algorithm for given images are evaluated. We will evaluate the SMG algorithm with the help of the image subset Size and using the area as object feature in Section 3.2.

3.1. Evaluation of the DPCS algorithm

The DPCS algorithm is a boundary based sequential technique. This procedure consists of several steps (Gerbrands, 1988). First, a Region Of Interest (ROI) is determined, which must enclose the whole object boundary under investigation. Then the part of image inside the ROI is re-sampled to a strip by using a polar transformation. In addition, a cost function related to the pixel edge values is defined on this strip. An optimal 8-neighbor connected curve can be found by using an optimization technique called dynamic programming (Martelli, 1976). This optimal curve is finally transformed back to the original image space as the required object boundary. curve is finally transformed back to the original image space as the required object boundary.

The DPCS algorithm will work automatically after finding the ROI. With synthetic images, the known location and size of objects can be used to accurately determine the ROI. To be consistent, a centered ROI limited by two circles (diameter 128 ± 101) is used for all evaluation experiments. Such a ROI can enclose the whole boundary of every object in the image subset Shape. We also use the best cost function discussed in (Gerbrands, 1988) for obtaining the cost images. Some segmentation results are shown in Fig. 2(a). The segmentation results become worse with the increase in noise level as expected.

The quantitative evaluation of this algorithm has been made according to Eq. (3). Corresponding to the use of the image subset Shape that is composed of images containing different elliptical objects, the object eccentricity is taken as the object feature in Eq. (3). The evaluation results presented by the curves of $RUMA_E$ against SNR are shown in Fig. 2(b) where a logarithm scale is used for both axes. The first observation from Fig. 2(b) is that the curve corresponding to $E = 2.5$ is quite distinct from others. This will be discussed in the next paragraph. The other curves have some comparable behavior. The quality of the final segmentation is fairly independent of the form of ROI as the same ROI is used for objects of different shape. This implies that the determination of ROI, in these situations, is not a critical factor affecting the performance of the DPCS algorithm. In contrast, the SNR level of images has more influence than that of object shapes over segmentation results. This has been shown by the rapid drop of the curves with the increase of the SNR levels.

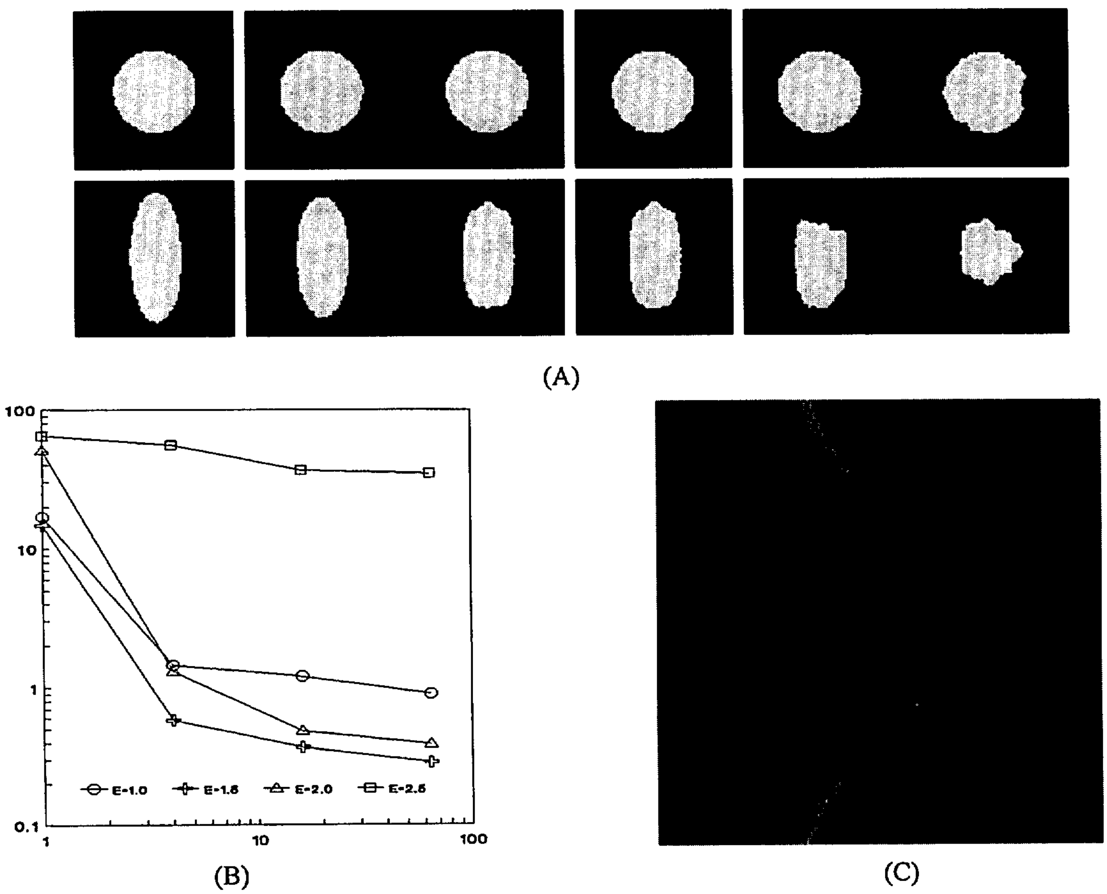


Fig. 2. Evaluation results of the DPCS algorithm. (a) Original and segmented objects: $E = 1.0$ (first row) and $E = 2.5$ (second row); (b) $RUMA_E$ vs. SNR; (C) A polar transformed cost image.

Now we discuss the problem associated with the curve $E = 2.5$. This could be explained more clearly in the transformed image space, in which the original ring enclosed by the ROI becomes a strip. One example is shown in Fig. 2(c), where the vertical axis is for the polar angle θ and the horizontal axis is for the vector from the re-sampling center ρ. In this figure, a portion (θ from 26° to 154°) of a cost image (obtained from one of $SNR = 64$ images) is presented together with the segmented boundary. The intensity of the image is proportional to the cost values. The top down black curve shows the boundary path actually found by the DPCS algorithm, which deviates from the higher cost region in the middle of the image (around $\theta = 90°$). It is in this part of boundary that some bigger segmentation errors have been produced. This problem can be analyzed as follows. The polar transformation is carried out in the original image space by re-sampling the ROI along the radius-vectors starting from the center of ROI. Each vector is re-sampled into one line in the transformed image. Because the DPCS algorithm searches precisely one pixel in each line of the cost image to form the optimal path and because this path should be an 8-neighbor connected one, a condition should be met to satisfy the connectivity constraint for this path. Suppose $\Delta\rho$ is the distance between the ends of two adjacent vectors ρ_θ and $\rho_{\theta+\Delta\theta}$ in the original image space, where $\Delta\theta$ is the sampling angle (the angle between two adjacent vectors), this condition can be represented by

$$\Delta\rho = |\rho_\theta - \rho_{\theta+\Delta\theta}| \leq 1. \tag{4}$$

Eq. (4) is a necessary condition that should be satisfied by a connected boundary path. In cases that the object boundary is irregular, Eq. (4) should be verified for every θ. For regular boundary, this problem can be solved by over-sampling.

In addition, we would like to point out that such a phenomenon related to the shape and curvature of object boundary can occur even for noise-free images. This can be easily seen from the results shown in the second column of Fig. 2(a). Finally, the relative flat curve corresponding to $E = 2.5$ in Fig. 2(b) indicates that the noise influence is minor if it is compared with that of the object shapes in cases where the elongation of objects exceeds certain limits.

3.2. Evaluation of the SMG algorithm

The SMG algorithm works by splitting the inhomogeneous regions of images into homogeneous components and merging or grouping the similar regions (Strasters and Gerbrands, 1991). As a split-merge based method (see, e.g., Horowitz and Pavlidis, 1976), it needs to pre-set several parameters according to the statistic properties of images. One critical parameter, when the (best) pseudo variance criterion is used as homogeneity criterion (Strasters and Gerbrands, 1991), is the Variance Within Homogeneous Regions (*VWHR*). It should be set equal to the Standard Deviation of Noise (*SDN*) of images for a proper segmentation. In real applications, the *SDN* is rarely known in advance. Besides, certain pre-processing techniques, such as low-pass filtering, can also alter the noise level in images. So, the *VWHR* can only be set according to some a priori knowledge about the image acquisition conditions and/or some rough estimations from images in practice. It is thus interesting to study the performance of the SMG algorithm when the *VWHR* was not set equal to the *SDN* of images.

The quantitative evaluation using the image subset Size is made according to Eq. (3) by taking the object Area as the object feature, i.e., $RUMA_A$ are computed. Since we know exactly the statistic properties of all test images, it is possible to precisely pre-set *VWHR* regarding the different noise levels in the images. Let Variance Ratio (*VR*) be the ration of *VWHR* to *SDN*. For each *SDN*, five *VWHR* values arechosen, which correspond to $VR = 0.5$, 0.75, 1.0, 1.25 and 1.5, respectively. Some segmented images obtained by these settings for images of $SDN = 8$ ($SNR = 16$) and $SDN = 32$ ($SNR = 1$) are shown in Fig. 3(a). These images contain objects with diameter 128. The evaluation results for images containing objects with diameters 128 and 64 are separately shown in Fig. 3, (b) and (c). In these two figures, the vertical axis presents the Accuracy Ratio (*AR*), which is the ratio of $RUMA_A$ obtained with various settings of *VWHR* to $RUMA_A$ obtained with the exact setting of *VWHR*. Here "exact setting" means that the value of *VWHR* is set equal to the *SDN* of images. When $VR = 1$, we have $AR = 1$. The bigger the *AR* values, the worse are the segmentation results (regarding to the feature used).

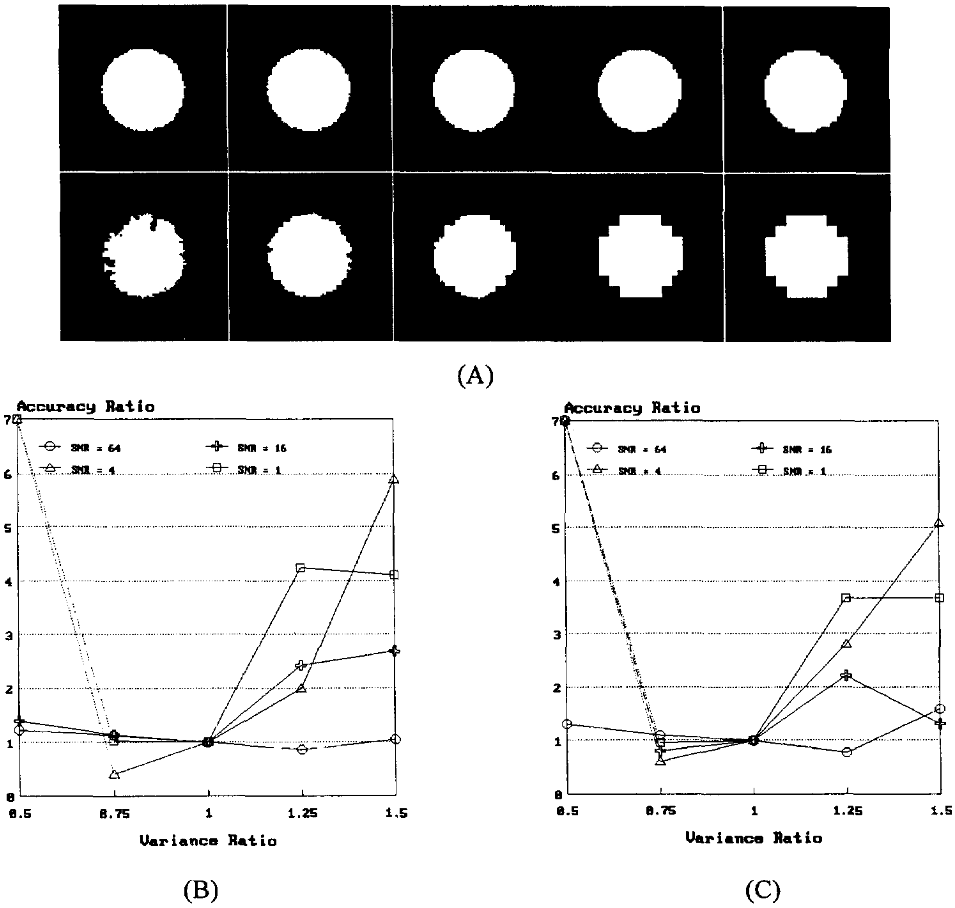


Fig. 3. Evaluation results of the SMG algorithm. (a) Segmented images containing object #8: $SNR = 16$ (first row) and $SNR = 1$ (second row); (b) Evaluation results (object #8); (c) Evaluation results (object #5).

In Fig. 3(a), the object contours become more regular as the VR increases and it shows more details as the VR decreases. The difference is more evident if one looks at the second row. Moreover, the segmentation results become worse if we look at images top down. This is no surprising as SDN gets bigger along this direction. Now let us analyze Fig. 3, (b) and (c). First, some similarities exist. When $VR \neq 1$, most AR values are quite bigger than 1. Note for $VR > 1$, the algorithm tends to under-segment images, while for VR, the algorithm tends toover-segment images and faces the problem of excessive computation cost. The valley in the middle of these curves indicates that both $VR > 1$ and VR settings affect the performance of the algorithm. With $VR = 0.5$, as most object and background regions have been merged, very large AR values are produced in consequence. The two figures have also some different points. A noticeable one is that in Fig. 3(b), only for the images with $SNR = 4$ and $SNR = 1$, the AR values corresponding to $VR = 0.5$ attain the maximum, while in Fig. 3(c), the maximum is also for the images with $SNR = 16$. This difference implies that the consequence of incorrect $VWHR$ setting is more serious with small objects. In other words, the correct setting of $VWHR$ is more important for segmenting images containing smaller objects.

This investigation shows that it is important to get VR around 1 to obtain good segmentation results. Other values of VR often produce reduced quality of segmented images, especially with lower *SNR* images. To a certain extent, using smaller *VR* values can provide slightly better results, but one takes the risk of getting more zig-zag and irregular object boundary (see Fig. 3(a)) and pays a lot of extra computations. The computing time for $VR = 0.75$ is about 10 times as for $VR = 1$ in our experiments. In conclusion, for a proper segmentation with the SMG algorithm,a critical task is to set the parameter *VWHR* appropriately. For this reason, an accurate estimation of the noise level from images might be essential for the proper operation of this algorithm.

4. Segmentation comparison

Now we proceed to the comparison of algorithms. We will first give a short description of the CEBC, IHCA and T-B algorithms, and then make the comparison of all algorithms mentioned is Section 2 with the aid of the synthetic image subset Size and by using $RUMA_A$ as the performance judge parameter.

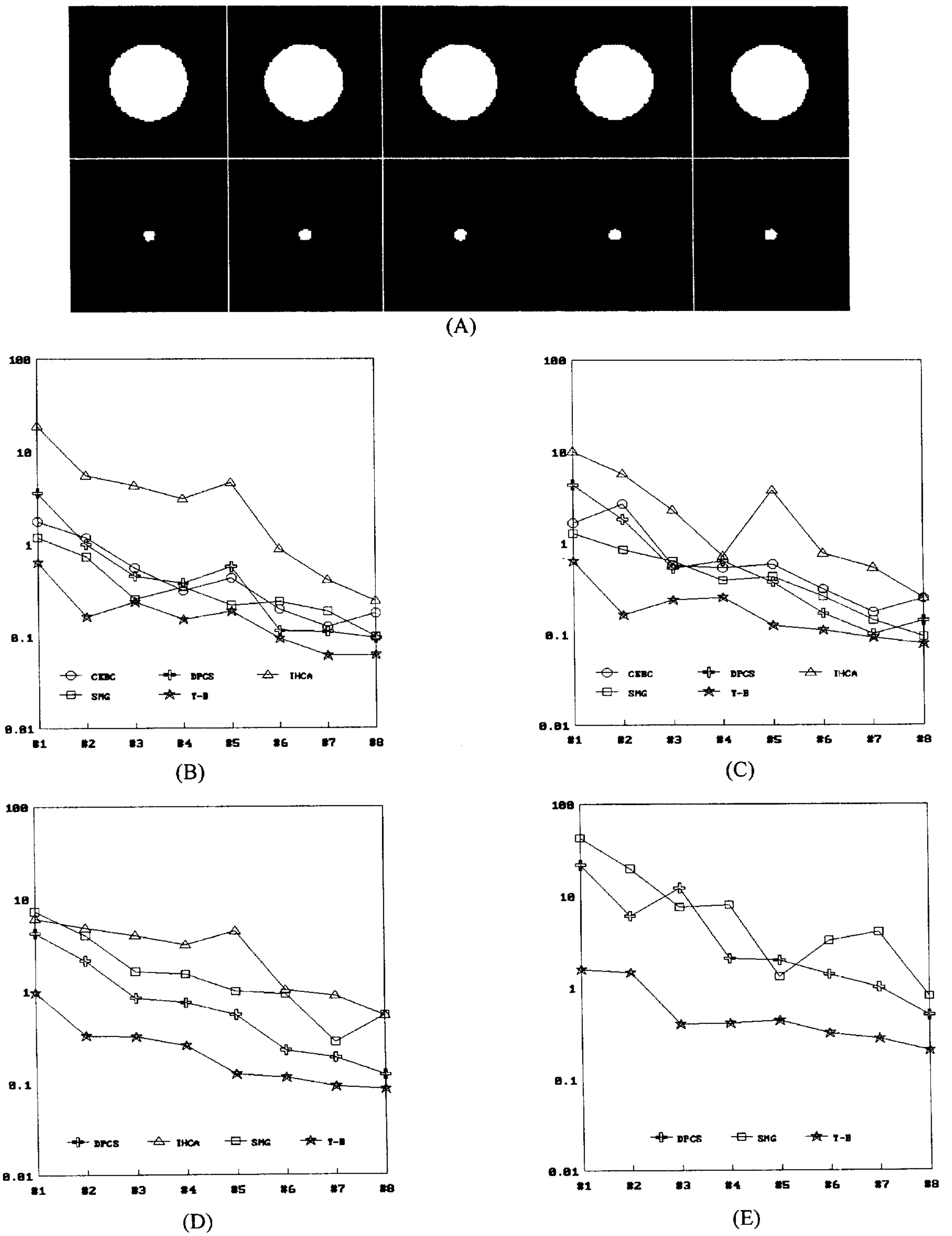
4.1. Specification and implementation of algorithms

The CEBC algorithm starts with detecting boundary edges using Canny operator (Canny, 1986). The Canny operator is a popularly employed edge detector that attempts to provide well located and thinned edge. In most cases, the detected edges are well located for our experiments. With noisy images, however, a closed edge boundary even for simple circular objects can not be guaranteed. In order to extract objects from images some further processes are needed to produce a closed boundary. The following simple morphology operations are used to keep the whole procedure quasi-parallel. After edge detection, all detected edge elements are first dilated two times to attempt to fill the gaps along the boundary, and then a skeleton process is performed on the dilated image. The largest component is selected from the skeleton image and is taken as the object boundary. By filling the inner region of this boundary, we obtain the required object. In our experience, this procedure can work well with images of $SNR = 64$ and $SNR = 16$. For images with even lower *SNR*, the edges detected by Canny operator are quite fragmented and dispersed. It is hard to find a closed boundary from such a result without more a priori knowledge.

The IHCA algorithm is an automatic thresholding technique with no free parameter. It searches the concavities of the gray level histogram of images to find good candidates for threshold values (Rosenfeld and Torre, 1983). It was initially proposed for 2-D image segmentation and has been extended to segment 3-D images with some improvements (Zhang et al., 1990). In a comparative study of five thresholding techniques made recently (Zhang and Gerbrands, 1992b) the IHCA algorithm has shown to be relatively suitable to segment images containing objects of different sizes. Its performance is more consistent than others with the variation of object size.

The implementations of the DPCS and SMG algorithms are just as those described in their evaluation. In this comparison, the parameters of SMG are all set in accordance with the test images (in particular, $VR = 1$). Such a setting in some senses should provide the best possible performance of this algorithm. This is also true for the DPCS algorithm as the ROI is determined by using the a priori knowledge about object size and location in images. Besides, a sampling rate that is high enough to prevent the problem discussed in Section 3.1 is used. From this point of view, the IHCA algorithm is in a sense anticipated by the DPCS and SMG algorithms, since similar information is hard to be incorporated into that algorithm even all characteristics of images are in hand. For a better comparative study, the best results that could be obtained by thresholding techniques are included under the name of T-B algorithm (as mentioned in Section 2.1). The T-B algorithm just uses all possible threshold values to segment images and takes the best results obtained according to the performance assessing criteria. This algorithm thus shows the best possible performance of all global

Fig. 4. Comparison results of the five different algorithms. (a) Segmented images containing object #8 (first row) and object #1 (second row); (b) Comparison results ($SNR = 64$); (c) Comparison results ($SNR = 16$); (d) Comparison results ($SNR = 4$); (e) Comparison results ($SNR = 1$).


(A)
100
10
1
0.1
0.01
#1 #2 #3 #4 #5 #6 #7 #8
CEBC
DPCS
IHCA
SMG
T-B
(B)
(C)
DPCS
IHCA
SMG
T-B
(D)
DPCS
SMG
T-B
(E)

thresholding algorithms because these algorithms are mainly different one from another by the methods for selecting an appropriate threshold value. Besides, the T-B algorithm can serve as a reference of limitation in the development of threshold selection methods. By comparing the performance of a thresholding algorithm with that of the T-B algorithm, we can know if this algorithm can be further improved or not.

The type, amount and manner of a priori knowledge incorporated into segmentation algorithms can greatly affect their performance (Zhang and Gerbrands, 1991). Although with synthetic images much a priori knowledge is available, the less such a priori knowledge was used, the more the algorithm's properties would be revealed from the performance study. On the other hand, it is desirable to use some similar a priori knowledge for all procedures in the comparison. In this study, the a priori knowledge that the object size is bigger than that of noise cluster is used. For example, the largest component in images is selected as the object with the SMG algorithm. For the T-B and IHCA algorithms, similar information has been used since after thresholding the biggest labeled region is selected. While for the CEBC and DPCS algorithms, such information is only implicitly used in the search of boundary.

4.2. Experimental results and discussions

The above five algorithms have been applied separately to segment the test images. Several segmented images obtained with these algorithms are shown in Fig. 4(a) as examples. The quantitative results of comparison are presented in Fig. 4, (b)–(e). In these figures, the horizontal axis is for the object label corresponding to object size, while the vertical axis is for $RUMA_A$ with logarithm scale. Since the CEBC algorithm has not been applied to images with $SNR = 4$ and $SNR = 1$, only in Fig. 4(b) and Fig. 4(c) its curves are presented. The IHCA algorithm does not provide acceptable results for images with $SNR = 1$, its curve has been omitted from Fig. 4(e).

The following observations and discussions can be made from Fig. 4:

(1) First let us look at Fig. 4(a). The five columns of images, from left to right, are images segmented with the CEBC, DPCS, IHCA, SMG and T-B algorithms, respectively. The object diameters in the first and second rows are 128 and 20, respectively. The difference among the segmented images in the second row is more visible than in the first row. However, the qualities of these images are not easy to be ranked visually. An objective assessment based on quantitative criteria is necessary to precisely judge these results.

(2) Now look at Fig. 4, (b)–(e). Due to the random nature of noise, some curves are not very smooth. Their trend of dropping with the increase of object size, however, is obvious. It shows that all algorithms perform better for images containing bigger objects than for images containing smaller objects. This confirms the statement that segmenting smaller objects is more difficult than segmenting bigger objects (Zhang and Gerbrands, 1992b). Such a difficulty is algorithm dependent, which has been manifested by the different inclinations of the curves corresponding to different algorithms.

(3) When the *SNR* level of images is higher, the CEBC algorithm shows some comparable performance with other algorithms. This was a little unexpected, but in retrospect, hardly surprising. With images having relatively higher *SNR*, the edge based methods should have the advantage to accurately locate the object boundary. Another good point of the CEBC algorithm is its parallelism which permits a fast implementation.

(4) The general performance ranking of the DPCS algorithm over the whole *SNR* range, especially with relatively lower *SNR* levels (see Fig. 4, (d) and (e)), is quite attractive. The DPCS algorithm is relatively robust in the presence of noise (Gerbrands, 1988). On the other side, the DPCS curves are more oblique than other curves in these figures. This is because its performance is more heavily affected by the change of object sizes in images.

(5) The IHCA algorithm performs worse than other algorithms for almost all test images, especially for images containing smaller objects. This is not very surprising. Histograms obtained from images with lower size ratio of object to background would be relatively flat. Searching the concavity in such histograms should be quite difficult. With the decrease of *SNR* level (till $SNR = 4$), however, the performance of IHCA becomes increasingly comparable with other algorithms as its curves become

closer and closer to other ones. This fact implies that the performance of IHCA deteriorates more slowly than other algorithms with the change of object sizes in images.

(6) For images containing smaller objects, the SMG algorithm ranks higher than the DPCS algorithm when the *SNR* level of images is higher (see Fig. 4, (b) and (c)). Its performance quickly declines as the *SNR* of images decreases and becomes worse than that of DPCS (see Fig. 4, (d) and (e)). In considering the performance robustness against the influence of noise, the SMG algorithm is worse than the DPCS algorithm though noise has an effect on both of them. In considering the performance consistency with the change of object sizes in images, however, the SMG algorithm is better than the DPCS algorithm.

(7) Finally, one can note from all these figures that the algorithm T-B shows the best performance among the five algorithms. This was somewhat unexpected and is quite interesting. It was a common belief that due to the parallel nature of thresholding processes, the performance of threshold selection algorithms is limited. In other words, if the image is severely contaminated by noise the thresholding techniques would give poor segmentation results. Here the experimental results provide some surprising pictures. If an appropriate threshold value can be obtained, the thresholding technique can still be very powerful even for segmenting images of wide SNR range. It seems that the relatively poorer performance of thresholding techniques (e.g., the IHCA algorithm in this study) with noisy images is mainly due to the inappropriate determination of threshold values.

5. Concluding remarks

An evaluation and comparison study of some typical segmentation algorithms from different groups is made with the aims of improving the performance of segmentation. Three particularities of this study are: (1) both segmentation evaluation and segmentation comparison are performed which revealed the behavior, advantages and limitations of these algorithms; (2) different types of algorithms are treated, which shows the generality of the approach; (3) carefully designed synthetic images are used for test experiments, so different image contents and imaging conditions can be studied. Though other algorithm classification schemes exist, the methodology presented in this study would be valid for treating all segmentation algorithms as no parameter of algorithms is required in evaluation and no restriction on the types of algorithms is imposed in comparison.

More than 1000 different segmentation algorithms have been proposed in the literature (Zhang and Gerbrands, 1994). Their evaluation and comparison would be a challenge research subject in segmentation. The most important is a standard procedure to combine the results of each new study with that of one made already. In this manner, a large number of algorithms can be studied and the results will be comparable. The procedure used in this study can fulfill these conditions. The comparison results can be directly connected to the results made from other studies using the same test images and the same performance criteria, for example, the comparison of different thresholding algorithms presented in (Zhang and Gerbrands, 1992b).

Practically, this study also provides some useful guidelines for the better use of these examined algorithms. For example, the evaluation results show how to avoid geometrical problem for segmenting elongated objects with the DPCS algorithm and indicate why should one put more attention on the appropriate setting of the parameter *VWHR* in the SMG algorithm. More generally, this study can be useful also for refining and improving other related segmentation algorithms.

Finally, there are some interesting findings from this study. For example, the best results are obtained by using the T-B algorithm and the performance of the IHCA algorithm is relatively inferior comparing with other algorithms. It appears that improving the threshold selection methods should be given more attention and would also be a promising research direction for developing new powerful segmentation techniques.

Acknowledgements

The support of the grants NNSF-69672029 and SCE-F1994660 is gratefully appreciated.

References

Canny, J., 1986. A computational approach to edge detection. IEEE Trans. Pattern Anal. Machine Intell. 8, 679–698.

Conzalez, R.C., Woods, R.E., 1992. Digital Image Processing, Addison-Wesley, Reading, MA.

Fu, K.S., Mei, J.K., 1981. A survey on image segmentation. Pattern Recognition 13, 3–16.

Gerbrands, J.J., 1988. Segmentation of Noisy Images. Delft University of Technology, The Netherlands.

Horowitz, S.L., Pavlidis, T., 1976. Picture segmentation by a tree traversal algorithm. J. ACM 23, 368–388.

Kitchen, L., Rosenfeld, A., 1981. Edge evaluation using local edge coherence. IEEE Trans. SMC 11, 597–605.

Lee, S.U. et al., 1990. A comparative performance study of several global thresholding techniques for segmentation. CVGIP 52, 171–190.

Martelli, A., 1976. An application of heuristic search methods to edge and contour detection. Comm. ACM 19, 73–83.

Pal, N.P., Bhandari, D., 1993. Image thresholding: Some new techniques. Signal Processing 33, 139–158.

Pal, N.R., Pal, S.K., 1993. A review on image segmentation techniques. Pattern Recognition 26, 1277–1294.

Rosenfeld, A., 1981. Image pattern recognition. Proc. IEEE 69, 596–605.

Rosenfeld, A., Torre, P., 1983. Histogram concavity analysis as an aid in threshold selection. IEEE Trans. SMC 13, 231–235.

Sahoo, P.K. et al., 1988. A survey of thresholding techniques. CVGIP 41, 233–260.

Strasters, K.C., Gerbrands, J.J., 1991. Three-dimensional image segmentation using a split, merge and group approach. Pattern Recognition Lett. 12, 307–325.

Zhang, Y.J. et al., 1990. Thresholding three-dimensional image. SPIE 1360, 1258–1269.

Zhang, Y.J., Gerbrands, J.J., 1991. Transition region determination based thresholding. Pattern Recognition Lett. 12, 13–23.

Zhang, Y.J., Gerbrands, J.J., 1992. On the design of test images for segmentation evaluation. In: Proc. EUROSCO, vol. 1, pp. 551–554.

Zhang, Y.J., Gerbrands, J.J., 1992. Comparison of thresholding techniques using synthetic images and ultimate measurement accuracy. In: Proc. 11th ICPR, vol. 3, pp. 209–213.

Zhang, Y.J., Gerbrands, J.J., 1994. Objective and quantitative segmentation evaluation and comparison. Signal Processing 39, 43–54.

Zhang, Y.J., 1996. A survey on evaluation methods for image segmentation. Pattern Recognition 29, 1335–1346.