

An Image Segmentation-Based Thresholding Method

Pei-Yan Pai

Department of Computer Science, National Tsing Hua University, No. 101, Section 2, Kuang-Fu Rd., Hsinchu 300, Taiwan, R.O.C.

Chin-Chen Chang

Department of Computer Science, National Tsing Hua University, No. 101, Section 2, Kuang-Fu Rd., Hsinchu 300, Taiwan, R.O.C.

Department of Information Engineering and Computer Science, Feng Chia University, No. 100, Wenhwa Rd., Seatwen, Taichung 407, Taiwan, R.O.C.

Yung-Kuan Chan and Meng-Hsiun Tsai

Department of Management Information Systems, National Chung Hsing University, No. 250, Kuokuang Rd., Taichung, Taiwan, R.O.C.

E-mail: ykchan@nchu.edu.tw

Shu-Wei Guo

Department of Computer Science and Engineering, National Chung Hsing University, No. 250, Kuokuang Rd., Taichung, Taiwan, R.O.C.

Abstract. Image segmentation is typically used to detect object contours in an image. In this article, a threshold-gradient based image segmentation method (TGISM) is proposed to extract objects from an image, based on the techniques of object contour gradient, gradient vector flow, and thresholding. In this article, a thresholding method minimizing within-class variance is also presented. The experimental results validate that the TGISM can give better segmentation results than the results obtained by the existing set of methods reported in the literature. In this article, a genetic algorithm based on accumulated historical data is proposed for determining the appropriate values of the parameters used in the TGISM. © 2012 Society for Imaging Science and Technology.

[DOI: 10.2352/J.ImagingSci.Technol.2012.56.3.030503]

INTRODUCTION

Object segmentation is the process of partitioning a digital image into multiple regions (sets of adjacent pixels). Each region can be considered as an object. The goal of segmentation is to simplify and/or change the representation of an image into objects that are more meaningful and easier to analyze. Thresholding is to determine the most suitable thresholds for separating a set of data into classes. Threshold-based segmentation is widely applied due to its simplicity and effectiveness.¹ Appropriate threshold values have to be carefully chosen to achieve the best results.

Many thresholding methods have been proposed to date. In-depth survey and evaluation of various thresholding methods are given by Refs. 2–7. Most of the existing

thresholding methods are based on the assumptions that the clusters in the pixel intensity histogram correspond to certain objects of interest. By separating these histogram clusters, objects of interest can be extracted. The histogram clustering offers a simple and practical option.

Otsu's thresholding method (OTM)¹ is one of the most commonly used thresholding methods. It selects an adaptive threshold by maximizing the between-class variances or minimizing the within-class variances of the data in the categorized classes. Refs. 4 and 6 concluded that the Otsu thresholding method is one of the best threshold selection methods for real world images with respect to uniformity and shape measures. However, if the variances or the quantities of the data among different classes are quite different, OTM cannot provide suitable thresholds effectively to classify the data into classes.¹

Hou et al.⁸ presented a minimum class variance thresholding method (MCVTM) to decide the most suitable thresholds based on the minimal class variances of the data in the categorized classes. However, if the variances of data between two classes are quite different, MCVTM cannot provide an ideal threshold to classify the data. Cheng et al.⁹ used the fuzzy c-partition algorithm and the entropy principle to select the thresholds. To obtain the best thresholds, the gray-level histogram of an image is partitioned into several sets by the fuzzy c-partition algorithm, which satisfies a maximum entropy restriction. The thresholds are the crossover points of the fuzzy sets that form the fuzzy partition.

Lee and Yang¹⁰ provided a multilevel histogram thresholding method to partition an image into different regions. However, this method is subject to many constraints on

Received Aug. 22, 2011; accepted for publication Jun. 18, 2012; published online Oct. 22, 2012.

1062-3701/2012/56(3)/030503/20/\$20.00

its applications, and it also requires manual intervention. Ng^{11,12} who proposed the valley-emphasis method (VEM) considered that the optimal thresholds for segmenting objects from an image generally lie near the wave valleys of the gray-level histogram of the image. However, not all the acceptable thresholds are at the lowest wave valleys in the gray-level histogram; hence, in many cases, VEM cannot determine an optimal threshold.

In this article, a minimal within-class standard deviation thresholding (MWCSDT) method is proposed to improve the effectiveness of OTM in determining proper thresholds by dividing class variance by class interval. The MWCSDT method is more versatile at selecting application-dependent thresholds by changing the values of two parameters, which describe the relative importance among data size, standard deviation, and class interval of a class.

The histogram does not convey sufficient information to discriminate image features; however, it can be complemented by other methods for dealing with complex segmentation tasks. In general, the pixels on the object contour have higher gradients. An active contour¹³ is used extensively in computer vision rendering and image processing, particularly for object segmentation. The threshold-gradient based image segmentation method (TGISM) hence integrates the MWCSDT method, the gradient of object contour, and the gradient vector flow-active contour model (GVF-ACM)¹³ to detect object contour.

TGISM can be applied to various applications of image segmentation. For a certain application, one can accumulate historical data. After that, for a specific application the accumulated historical data can be applied to train and determine the most appropriate values of the parameters that are used in TGISM. In this article, a genetic algorithm (GA) is proposed to determine the best values of the thresholds from the accumulated historical data.

The remainder of the article is organized as follows. The “Related works” section briefly reviews some techniques that are used in this article. The “Threshold-gradient based image segmentation method (TGISM)” section describes the proposed method in detail. The “Experiments” section shows the experimental results. Some conclusions are given in the last section.

RELATED WORKS

Techniques used in this article are briefly reviewed in this section.

Otsu's Thresholding Method

Let I be a gray-level image, the gray levels of which are distributed between 0 and $L-1$, where L is the maximal gray level in the image. N_l is the number of pixels with gray level l , and N is the total number of pixels in I . The probability of a pixel with gray level l in I (denoted as p_l) can be computed as

$$p_l = \frac{N_l}{N}. \quad (1)$$

The average gray-level M of I is

$$M = \sum_{l=0}^{L-1} lp_l. \quad (2)$$

To partition a set of data into two classes, all the pixels in I are separated into two classes C_1 and C_2 by using a threshold t , so that the gray levels of all the pixels in C_1 are less than t and those in C_2 are equal to or greater than t . The probabilities P_{C_1} and P_{C_2} of the pixels in C_1 and C_2 are defined as:

$$P_{C_1}(t) = \sum_{l=0}^{t-1} p_l \quad \text{and} \\ P_{C_2}(t) = \sum_{l=t}^{L-1} p_l, \text{ respectively.} \quad (3)$$

The average gray levels M_{C_1} and M_{C_2} of C_1 and C_2 are defined by

$$M_{C_1}(t) = \frac{\sum_{l=0}^{t-1} lp_l}{P_{C_1}(t)} \quad \text{and} \\ M_{C_2}(t) = \frac{\sum_{l=t}^{L-1} lp_l}{P_{C_2}(t)}, \text{ respectively.} \quad (4)$$

OTM then computes the optimal threshold t^* by minimizing the within-class variance, i.e.

$$t^* = \text{Arg}(\text{Min}_{0 \leq t < L} \{\sigma_{\text{Within}}^2(t)\}), \quad (5)$$

where σ_{Within}^2 is a within-class variance that can be computed as

$$\sigma_{\text{Within}}^2(t) = P_{C_1}(t) \times \frac{\sum_{l=0}^{t-1} N_l \times (l - M_{C_1}(t))^2}{N_{C_1}} + P_{C_2}(t) \\ \times \frac{\sum_{l=t}^{L-1} N_l \times (l - M_{C_2}(t))^2}{N_{C_2}}, \quad (6)$$

where N_{C_i} is the number of pixels in class C_i .

OTM is unable to determine a proper threshold when the variances or the number of data between classes are quite different. Figure 1(a) is a synthesized image consisting of two objects. Fig. 1(b) is the gray-level histogram of the image. Fig. 1(c) shows the segmentation result based on the threshold $t^* = 36$ obtained by OTM. Fig. 1(d) shows the target segmentation result from the image in Fig. 1(a). Since the variance and number of data in the class with darker pixels are much larger than those in the class with brighter pixels, OTM cannot obtain a good threshold to segment the object from the image in Fig. 1(a).

GVF-ACM

The active contour model (ACM),^{13,14} also called italic, evolves an initial contour towards final convergence under

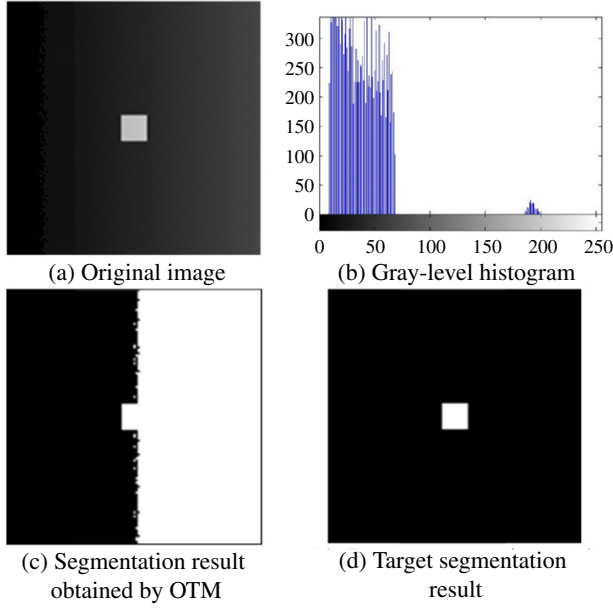


Figure 1. A gray-level image and the segmentation result by OTM.

the influence of both internal forces and boundary-regional image forces. Active contours are curves defined within an image domain that can move under the influence of internal forces coming from within the curve itself and external forces computed from the image data. The internal and external forces are defined so that the snake will conform to an object boundary within an image I . The active contour model is defined as an energy minimizing spline—the snake's energy depends on its shape and location within I . The energy function, which is minimized, is a weighted combination of internal and external forces. The internal forces emanate from the shape of the snake, while the external forces come from I and/or from higher-level, image-understanding processes.

The snake is parametrically defined as $c(s)(x(s), y(s))$, where $x(s)$ and $y(s)$ are x and y coordinates along the contour, and s is from $[0, 1]$. The energy function E to be minimized is $E = E_{\text{int}} + E_{\text{ext}}$. The internal spline energy E_{int} , including the first- and second-order differentiations of the contour $c(s)$, can be written as:

$$E_{\text{int}} = \int_0^1 \left(\frac{\alpha(s) |c_s(s)|^2 + \beta(s) |c_{ss}(s)|^2}{2} \right) ds, \quad (7)$$

where $\alpha(s)$ and $\beta(s)$ specify the elasticity and stiffness of the contour, respectively, and $c_s(s)$ and $c_{ss}(s)$ are the first- and second-order differentiations of $c(s)$, respectively. The image force or external force $E_{\text{ext}}(x) = -|\nabla [G_\sigma(x, y) * I(x, y)]|^2$ is to detect the specific image features. $[G_\sigma(x, y)]$ is a two-dimensional Gaussian function with standard deviation σ , $*$ is the convolution operator, and ∇ is the gradient operator.

There are two key problems with ACM. First, if the internal contour is not close to the true boundary, it will likely converge to the wrong result. Second, it is difficult

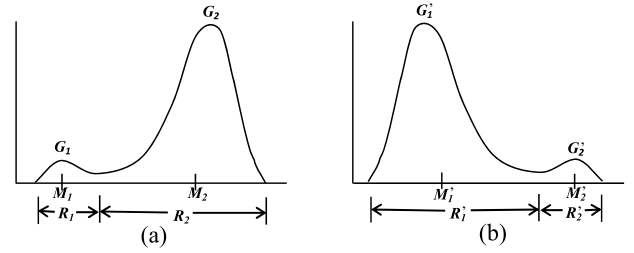


Figure 2. Two gray-level histograms with the same standard deviation but different mean values.

for active contours to progress into boundary concavities.¹³ Gradient vector flow (GVF) fields are provided to solve both of the problems specified above.¹³ GVF fields are dense vector fields derived from images by minimizing a certain energy function.

The GVF force combines optical flow with the gray-scale gradient. Let $f(x, y) = -|\nabla [G_\sigma(x, y) * I(x, y)]|^2$ be the edge map derived from I . $V(x, y) = (u(x, y), v(x, y))$ is the flow vector obtained by using two diffusion-based partial differential equations, $\mu \nabla^2 u - (u - f_x)(f_x^2 + f_y^2) = 0$ and $\mu \nabla^2 v - (v - f_y)(f_x^2 + f_y^2) = 0$, in the entire image domain, where μ is a weighting factor that determines the relative significance of the first and second terms, and ∇^2 is the Laplacian operator. In the GVF method, a moving parametric contour is driven by the minimization of energy

$$E_{\text{GVF}} = \iint \mu |\nabla V|^2 + |\nabla f|^2 |V - \nabla f|^2 dx dy. \quad (8)$$

Generally, for an image with more noise, a greater value is given to μ . Then the external force E_{ext} can be modified into $E_{\text{ext}} = \kappa E_{\text{GVF}}$, where κ is the GVF weight.

Segmentation evaluation

Misclassification error (ME),⁵ relative foreground area error (RAE),¹ and modified Hausdorff distance (MHD)⁵ are three frequently used segmentation error measures. For separating objects from image background, ME can be simply expressed as:

$$\text{ME} = 1 - \frac{|B_O \cap B_T| + |F_O \cap F_T|}{|B_O| + |F_O|}, \quad (9)$$

where B_O and F_O are the pixels of the expected background and foreground in the original image, B_T and F_T are the pixels of the segmented background and foreground in the image, $\cap$ is the intersection operator, and $|B_O|$, $|F_O|$, $|B_T|$ and $|F_T|$ are respectively the numbers of pixels in B_O , F_O , B_T , and F_T .

RAE is defined as

$$\text{RAE} = \begin{cases} \frac{A_O - A_T}{A_O}, & \text{if } A_T < A_O \\ \frac{A_T - A_O}{A_T}, & \text{if } A_T \geq A_O \end{cases} \quad (10)$$

where A_O is the areas of the expected objects and A_T is the areas of the segmented objects. MHD can be computed by:

$$\text{MHD}(C_O, C_T) = \frac{1}{|C_O|} \sum_{c_O \in C_O} \min_{c_T \in C_T} \|c_O - c_T\|, \quad (11)$$

where $\|c_O - c_T\|$ is the spacial (Euclidean) distance between two pixels c_O and c_T on the contours of the expected objects and on the contours of the segmented objects.

THRESHOLD-GRADIENT BASED IMAGE SEGMENTATION METHOD (TGISM)

In this article, a threshold-gradient based image segmentation method (TGISM) is proposed to detect the contours of objects in an image based on the minimal within-class standard deviation thresholding (MWCSDT) method, the gradient of object contour, and GVF-ACM.¹³ This section will describe them in detail.

Minimal within-class standard deviation thresholding (MWCSDT) method

In image segmentation, the gray levels of the pixels within a same object are often highly similar but quite different between distinct objects. Variance or standard deviation is often used to identify the discrepancy among the data in the same class. OTM, VEM, and MCVTM all categorize data according to the within-class variances, which should be as small as possible. Furthermore, via the number of data in each class, OTM and VEM combine the variances of data in all the divided classes into one variance to indicate the discrepancy of all the data within classes. In this article, we also used the variance and the number of the data in each class to compute the optimal thresholds.

When the means of the data in different data classes are significantly different, the standard deviations of the data cannot be compared directly. Hence, the standard deviation of the data in one class is often divided by the mean of the data in the same class for standardization. In Figure 2(a) is the histogram of the data set containing two data classes with standard deviations Std_1 and Std_2 as well as data means M_1 and M_2 . In Fig. 2(b) is the histogram of the data set containing two different data classes with standard deviations Std'_1 and Std'_2 as well as data means M'_1 and M'_2 . The data distributions in groups G_1 and G'_2 are the same, and the data distributions in groups G_2 and G'_1 are the same; that is $Std_1 = Std'_2$, $Std_2 = Std'_1$, $R_1 = R'_2$, and $R_2 = R'_1$. However, $M_1 \neq M'_2$ and $M_2 \neq M'_1$. In this article, we will normalize the standard deviation by dividing it by its corresponding class interval R .

Let $x_{\min}$ and $x_{\max}$ be the minimal and maximal pixel values, respectively, in a set of data that is to be separated into G classes according to their values. $G - 1$ thresholds $t_1, t_2, \dots, G - 1$ have to be specified, so that all the data in the g th class are in the interval between t_{g-1} and t_g . Let $x_{g,i}$ be the i th smallest value in class g , and $n_{g,i}$ be the number of data in class g with value $x_{g,i}$. Given a threshold set $t = (t_1, t_2, \dots, t_{G-1})$, the class interval $R_g(t)$ of class g is

-1	0	$+1$
-2	0	$+2$
-1	0	$+1$
G_x		

$+1$	$+2$	$+1$
0	0	0
-1	-2	-1
G_y		

Figure 3. Sobel convolution masks G_x and G_y in the x- and y-directions.

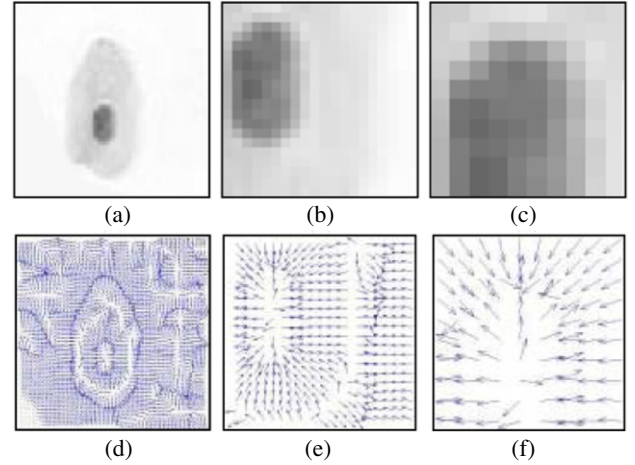


Figure 4. (a) Original image; (b) partial image of (a) in high resolution; (c) partial image of (b) in a higher resolution; (d)–(f) GVFs of the images in (a)–(c), respectively.

defined as

$$R_g(t) = \begin{cases} t_1 - x_{\min}, & \text{if } g = 1. \\ t_g - t_{g-1}, & \text{if } 1 < g < G. \\ x_{\max} - t_{G-1}, & \text{if } g = G. \end{cases} \quad (12)$$

The probability $P_g(t)$ is defined as the percentage ratio of data size in class g to the data size of the entire data set, i.e.

$$P_g(t) = \frac{\sum_{i=1}^{R_g(t)} n_{g,i}}{\sum_{g=1}^G \sum_{i=1}^{R_g(t)} n_{g,i}}. \quad (13)$$

The average gray level M_g of the pixels in the g th class is:

$$M_g = \frac{\sum_{i=1}^{R_g(t)} n_{g,i} x_{g,i}}{\sum_{i=1}^{R_g(t)} n_{g,i}}. \quad (14)$$

The standard deviation $Std_g(t)$ of the gray levels in the g th class is:

$$Std_g(t) = \sqrt{\frac{\sum_{i=1}^{R_g(t)} n_{g,i} (x_{g,i} - M_g)^2}{\sum_{i=1}^{R_g(t)} n_{g,i}}}. \quad (15)$$

The expected within-class standard deviation is defined by the following formula:

$$\sum_{g=1}^G \frac{P_g(t) Std_g(t)^{r_1}}{R_g(t)^{r_2}}, \quad (16)$$

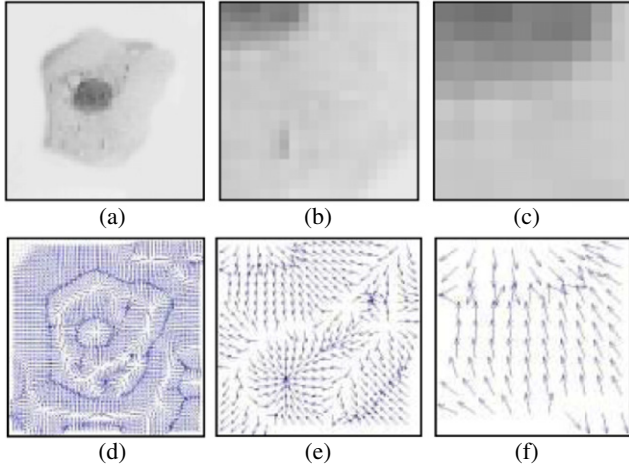


Figure 5. (a) Another image; (b) partial image of (a) in high resolution; (c) partial image of (b) in higher resolution; (d)–(f) GVFs of the images in (a)–(c), respectively.

where r_1 and r_2 are the weighting factors that determine the relative significance of R_g , P_g , and Std_g .

Gradient of object contour

An edge generally corresponds to a set of strong illumination gradients. The gradients can be obtained by calculating the derivatives of the image. Many gradient-computing methods, such as the widely used gradient, Roberts cross, and Sobel operators, have been proposed.^{4,5} Gradient and Roberts cross operations can generate fewer false edges, but they often give very weak edge gradients that make further segmentation difficult. In this article, the Sobel operation is adopted to calculate the gradients of the pixels in an image I . The Sobel operation performs a 2-D spatial gradient measurement on I and emphasizes the regions of high spatial gradients that correspond to edges. Two 3×3 convolution masks G_x and G_y , shown in Figure 3, are used in the Sobel operation.

Let $p_{i,j}$ be a pixel located at the coordinates (i, j) in I and $W_{i,j}$ be the corresponding window of $p_{i,j}$, which is at the center of $W_{i,j}$. $W_{i,j}$ consists of 3×3 pixels. The gradient $g_{i,j}$ of $p_{i,j}$ is defined as:

$$g_{i,j} = \sqrt{(G_x \otimes W_{i,j})^2 + (G_y \otimes W_{i,j})^2}, \quad (17)$$

where $\otimes$ is the convolution operator.

Let Min_g and Max_g be the minimal and maximal gradients of all the pixels in I , respectively. This approach assigns the value $\frac{g_{i,j} - \text{Min}_g}{\text{Max}_g - \text{Min}_g} \times 255$ to the gray level of the pixel located at the coordinates (i, j) in I_g to generate an image I_g . Hence, I_g can be used to describe the gradients of the pixels in I .

Gradient vector flow (GVF) feature

GVF-ACM¹³ is computed as a diffusion of the gradient vectors of a gray-level or binary edge map image. Figures 4 and 5 show the GVFs of two images in different resolutions.

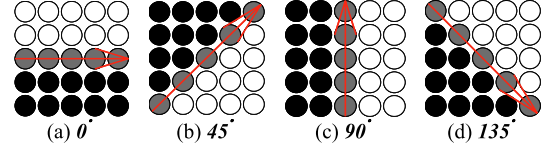


Figure 6. Four different partitions for $m_M = 5$.

In Fig. 4(d)–(f) and 5(d)–(f), an arrow signifies the GVF's orientation of a pixel. One could imagine that the edge on an image is the valley that the GVFs of its neighbors flow into. Moreover, the orientations of GVFs in the vicinity of the edge pixels are almost perpendicular to the edge. While the pixels lie between two edges, the GVFs in the neighborhood of the pixels stream from the pixels toward both edges. Therefore, the GVF of a pixel p_{ij} can be used to describe the possibility of p_{ij} being an edge pixel. Hence, in this article, the GVF feature is used to represent the possibility that a pixel is an edge pixel.

To calculate the GVF feature of a pixel p_{ij} in an image I , we assume that there is an edge passing through p_{ij} in I . First, the orientation of the edge is estimated, which is assumed to be close to one of 0° , 45° , 90° , or 135° , based on the corresponding window W_{ij} of p_{ij} in I , where W_{ij} consists of $m_M \times m_M$ pixels, and p_{ij} is at the center of W_{ij} . W_{ij} is divided into two equal regions according to the four different angles of 0° , 45° , 90° , and 135° . Figure 6 shows the four different partitions where $m_M = 5$. The black dots and the white dots signify the shadow (black) region and the highlight (white) region, respectively. We call the partitions in Fig. 6(a)–(d) θ -partitions for $\theta = 0^\circ$, 45° , 90° , and 135° , respectively. For each θ -partition, the average gray levels c_b and c_w of the black regions, respectively, and the white regions, as well as $d_\theta = |c_b - c_w|$, are calculated. Here, θ_e is the estimated direction of the edge, where $\theta_e = \arg(\max_{\theta=0^\circ, 45^\circ, 90^\circ, 135^\circ} (d_\theta))$.

Let θ_b and θ_w be the average directions of the GVFs of the pixels in the black region and white region, respectively. The GVF feature GVF_{ij} of p_{ij} is defined as

$$GVF_{ij} = \frac{\sin(\theta_b - \theta_e) + \sin(\theta_e - \theta_w) + 2}{4} \times 255. \quad (18)$$

The larger the value of GVF_{ij} , the greater the possibility that $p_{i,j}$ is an edge pixel. All of the GVF_{ij} values in I can construct an image I_{GVF} . Figure 7 displays some examples, showing the GVF features in different GVFs, and Figure 8(c) demonstrates the I_{GVF} of two images.

Feature combination thresholding

The TGISM integrates the minimizing within-class standard deviation, gradient of object contour, and GVF features to determine the best threshold for image segmentation. Let I be an original image and $I(i, j)$ be the pixel located at (i, j) in I . Let g_{ij} located at (i, j) in I_g be the gradient of pixel $I(i, j)$ and $G_{\max}$ be the maximal gray level of all the pixels in I_g , where $I_g(i, j)$ is the gradient of $I(i, j)$. Similarly, let GVF_{ij} be the GVF feature of $I(i, j)$ and $GVF_{\max}$ be the maximum of all the GVF_{ij} 's in I . Fig. 8(c) shows that the GVF feature can highlight object contour. However, it also

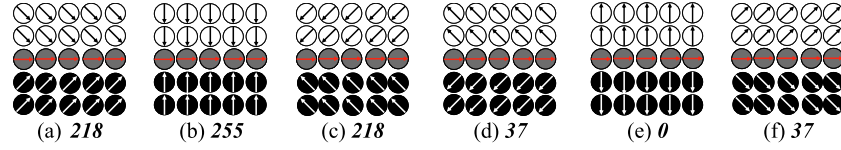


Figure 7. Examples that show differences in distinct GVFs.

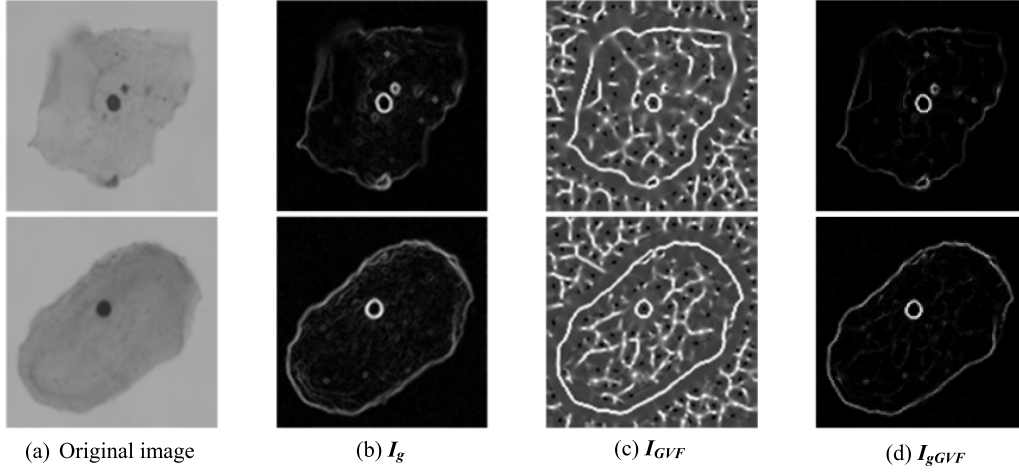
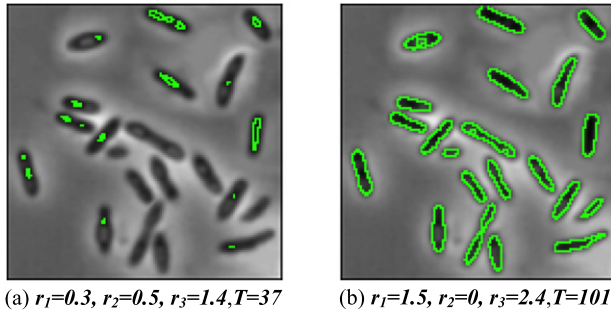

 Figure 8. I_g , I_{GVF} , and I_{gGVF} of two cervical smear images.

Table 1. Average ME, RAE, and MHD values obtained in Experiment 1.

Measure	TGISM	OTM	VEM	MCVTM
ME	0.00032	0.10053	0.08853	0.00051
RAE	0.00513	0.57625	0.56375	0.05428
MHD	0.00694	0.10603	0.10427	0.01025


 Figure 9. Segmentation results obtained by the TGISM with different r_1 , r_2 , and r_3 .

generates a significant amount of noise. Hence, the TGISM creates an image I_{gGVF} by combining I_g with I_{GVF} to enhance the gradient of object contour and to inhibit the associated noise. $I_{gGVF_{ij}}$, located at (i, j) in I_{gGVF} , is defined as

$$I_{gGVF_{ij}} = \frac{I_g(i, j)}{G_{\max}} \times \frac{GVF_{ij}}{GVF_{\max}} \times 255. \quad (19)$$

Fig. 8(d) illustrates the I_{gGVF} of the two cervical smear images in Fig. 8(a). It is easily observed that the gradients of most contour pixels are intensified and that the associated noise is suppressed. The TGISM uses I_{gGVF} to determine the appropriate threshold.

To separate the data into G classes by a given threshold set $t^* = (t_1, t_2, \dots, t_{G-1})$, for each threshold t_g , the TGISM computes $gGVF_g(t_g)$ and w_g using the following program segment:

For each pixel p_{ij} in I
 if $I_g(i, j) = t_g$ then
 $gGVF_g(t_g) = gGVF_g(t_g) + I_{gGVF_{ij}}$
 $count_g = count_g + 1$
 $gGVF_g(t_g) = \frac{gGVF_g(t_g)}{count_g}$
 For each $w_g = \frac{count_g}{\sum_{g=1}^{G-1} count_g}$.

For each possible threshold set $t = (t_1, t_2, \dots, t_{G-1})$, the TGISM computes the appropriate threshold t^* based on the following formula:

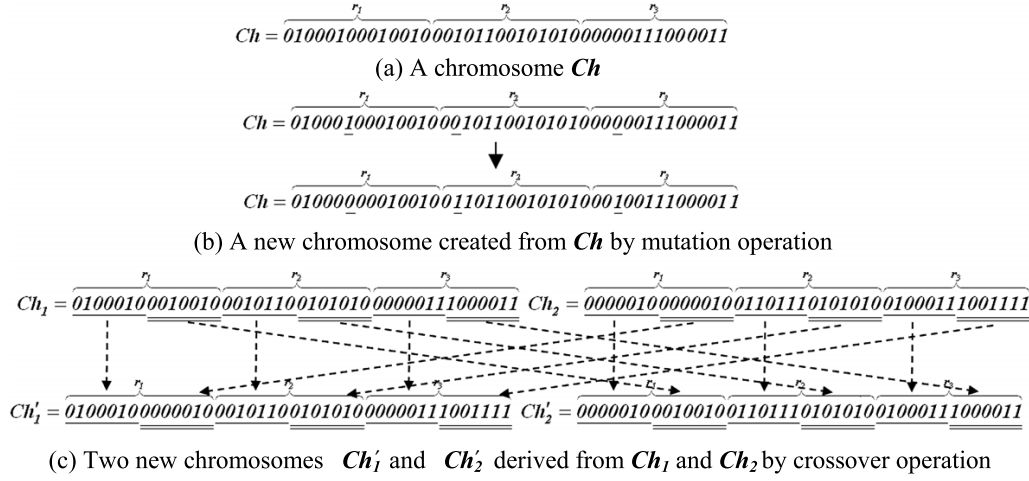


Figure 10. An example of the GBPD.

No	Original image	Target contour	TGISM	OTM	VEM	MCVTM
1						
			$t^*=(53)$	$t^*=(105)$	$t^*=(87)$	$t^*=(69)$
2						
			$t^*=(179)$	$t^*=(211)$	$t^*=(183)$	$t^*=(189)$
3						
			$t^*=(148)$	$t^*=(184)$	$t^*=(150)$	$t^*=(158)$
4						
			$t^*=(135)$	$t^*=(186)$	$t^*=(152)$	$t^*=(150)$

Figure 11. Four images, target contours, and detected contours.

$$t^* = \text{Arg} \left(\min_{t_g} \left(\frac{\sum_{g=1}^G \left(\frac{P_g(t) \text{Std}_g(t)^{r_1}}{R_g(t)^{r_2}} \right)}{\left(\sum_{g=1}^{G-1} (w_g \text{GVF}_g(t_g)) \right)^{r_3}} \right) \right), \quad (20)$$

where r_1 , r_2 , and r_3 are the weighting factors that determine the relative significance of R_g , P_g , Std_g , and GVF features.

The GBPD

Figure 9 shows that the segmentation results obtained by the TGISM are deeply affected by the values of r_1 , r_2 , and r_3 in Eq. (20). Genetic algorithms^{15,16} are heuristic search methods that mimic the process of natural evolution. In this article, a genetic algorithm will be employed to compute the

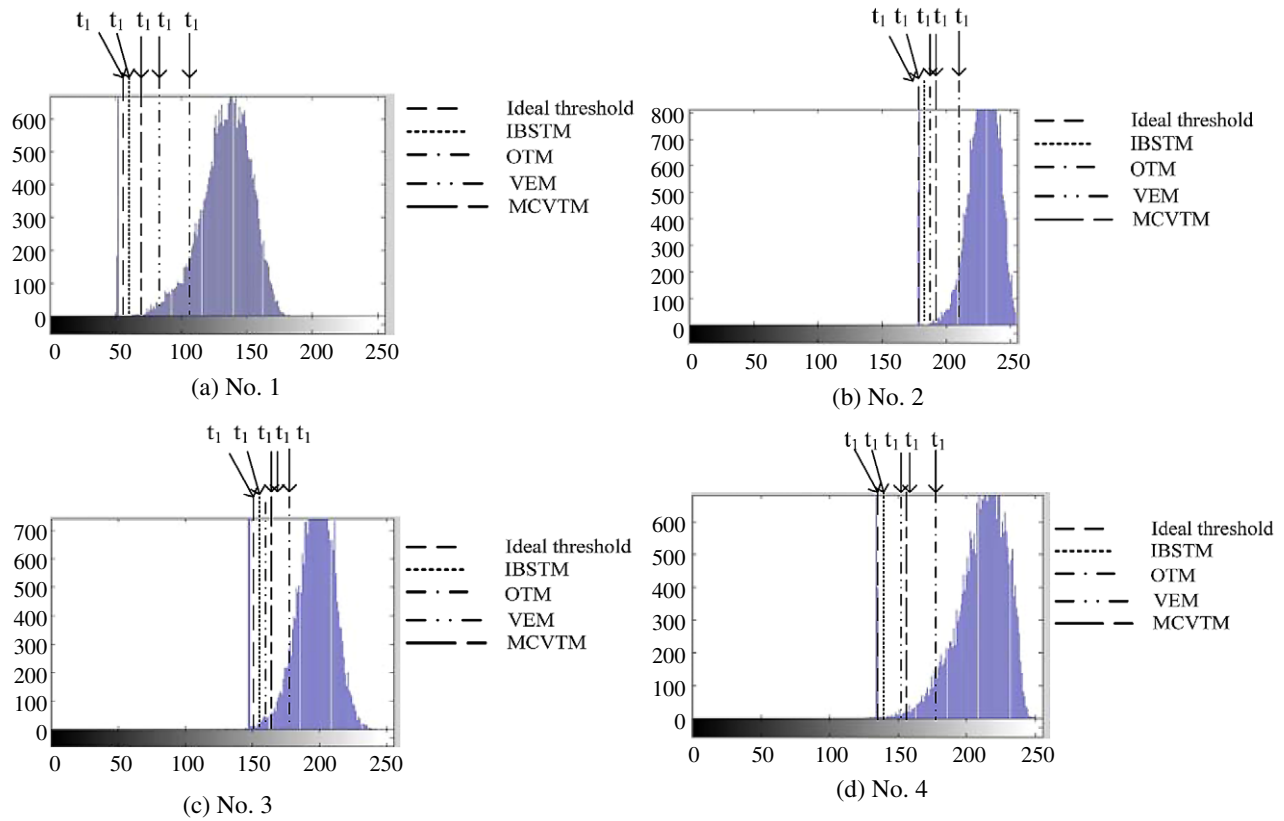


Figure 12. Gray-level histograms of test images in Fig. 11.

No	Original image	Target contour	TGISM	OTM	VEM	MCVTM
1						
			$T=119$	$T=125$	$T=128$	$T=138$
2						
			$t^*=(101)$	$t^*=(99)$	$t^*=(101)$	$t^*=(65)$
3						
			$t^*=(63)$	$t^*=(72)$	$t^*=(70)$	$t^*=(224)$

Figure 13. Three images, target contours, and contours, cut by TGISM, OTM, and VEM.

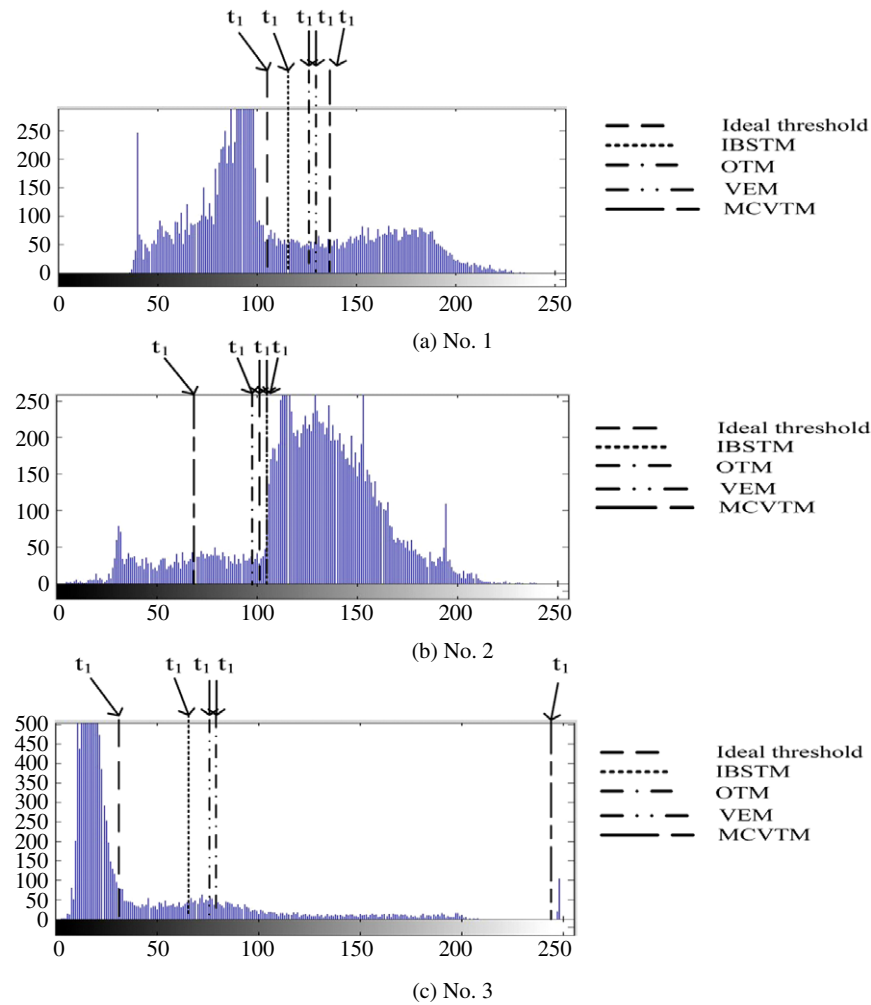


Figure 14. Gray-level histograms of the test images in Fig. 13.

Table II. Average ME, RAE, and MHD values obtained in Experiment 2.

Measure	TGISM	OTM	VEM	MCVTM
ME	0.11684	0.12217	0.12163	0.25135
RAE	0.17833	0.18169	0.20043	0.31776
MHD	2.04407	2.18067	2.21086	3.33966

Table III. Average ME, RAE, and MHD values obtained in Experiment 3.

Measure	TGISM	OTM	VEM	MCVTM
ME	0.01514	0.03133	0.03052	0.03802
RAE	0.02518	0.06259	0.06494	0.09174
MHD	4.86797	4.95207	4.98055	5.17541

most suitable values of r_1 , r_2 , and r_3 in Eq. (20). We will call it a genetic-based parameter detector (GBPD). The GBPD makes use of a binary string, concatenating three substrings s_1 , s_2 , and s_3 , to represent a chromosome Ch . s_1 , s_2 , and s_3 are corresponding binary string representations of r_1 , r_2 , and r_3 , respectively. For each chromosome Ch , r_1 , r_2 , and r_3 are

encoded as:

$$\begin{cases} r_1 = gap_1 \times n_1, \\ r_2 = gap_2 \times n_2, \\ r_3 = gap_3 \times n_3, \end{cases} \quad (21)$$

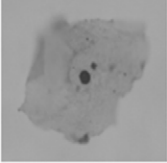
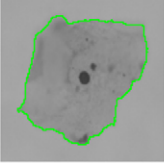
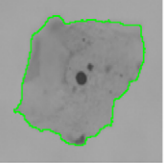
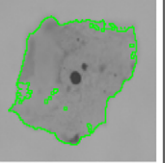
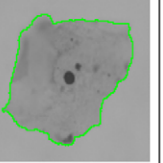
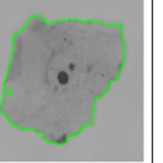
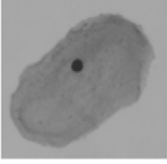
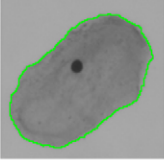
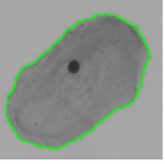
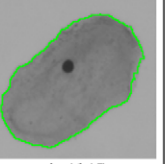
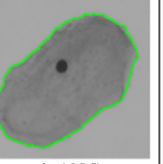
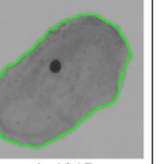
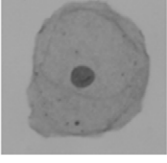
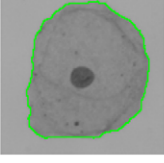
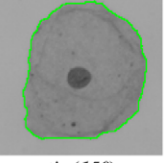
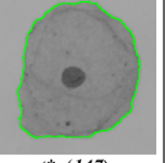
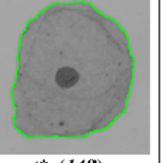
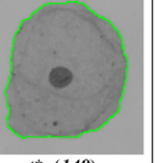
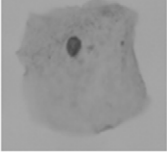
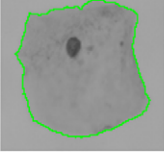
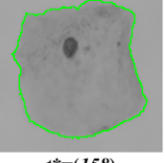
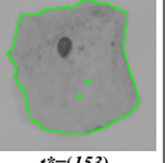
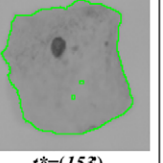
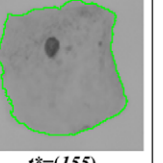
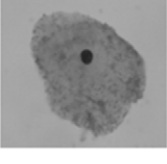
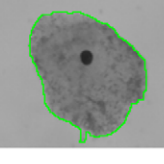
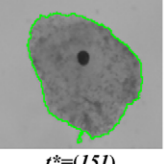
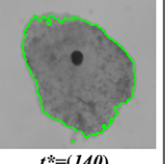
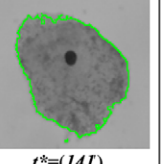
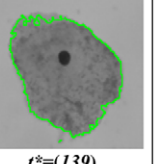
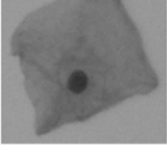
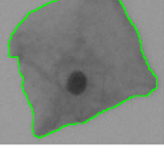
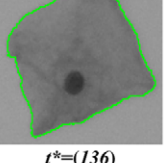
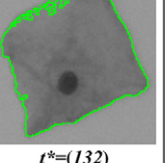
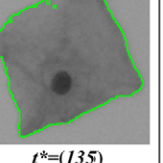
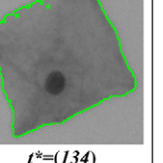
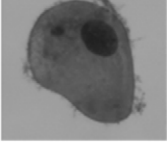
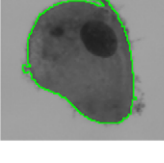
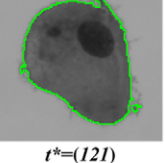
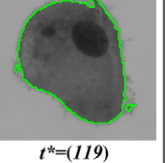
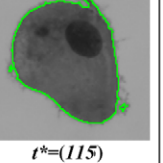
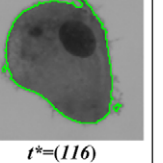
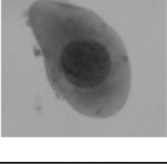
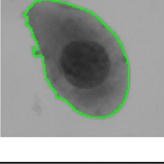
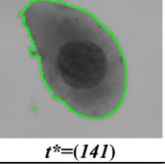
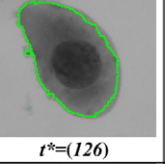
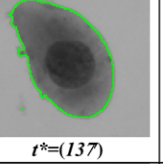
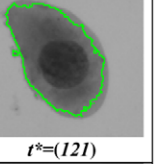
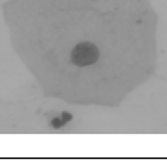
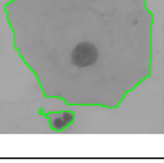
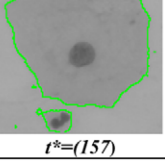
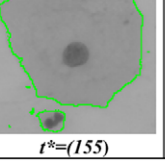
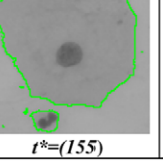
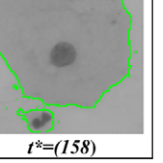
No	Original image	Target contour	TGISM	OTM	VEM	MCVTM
1			 $t^*=(155)$	 $t^*=(151)$	 $t^*=(158)$	 $t^*=(154)$
2			 $t^*=(151)$	 $t^*=(145)$	 $t^*=(151)$	 $t^*=(145)$
3			 $t^*=(150)$	 $t^*=(147)$	 $t^*=(148)$	 $t^*=(149)$
4			 $t^*=(158)$	 $t^*=(153)$	 $t^*=(153)$	 $t^*=(155)$
5			 $t^*=(151)$	 $t^*=(140)$	 $t^*=(141)$	 $t^*=(139)$
6			 $t^*=(136)$	 $t^*=(132)$	 $t^*=(135)$	 $t^*=(134)$
7			 $t^*=(121)$	 $t^*=(119)$	 $t^*=(115)$	 $t^*=(116)$
8			 $t^*=(141)$	 $t^*=(126)$	 $t^*=(137)$	 $t^*=(121)$
9			 $t^*=(157)$	 $t^*=(155)$	 $t^*=(155)$	 $t^*=(158)$

Figure 15. Twelve cervical smear images, the target contours, and the contours cut by TGISM, OTM, VEM, and MCVTTM.

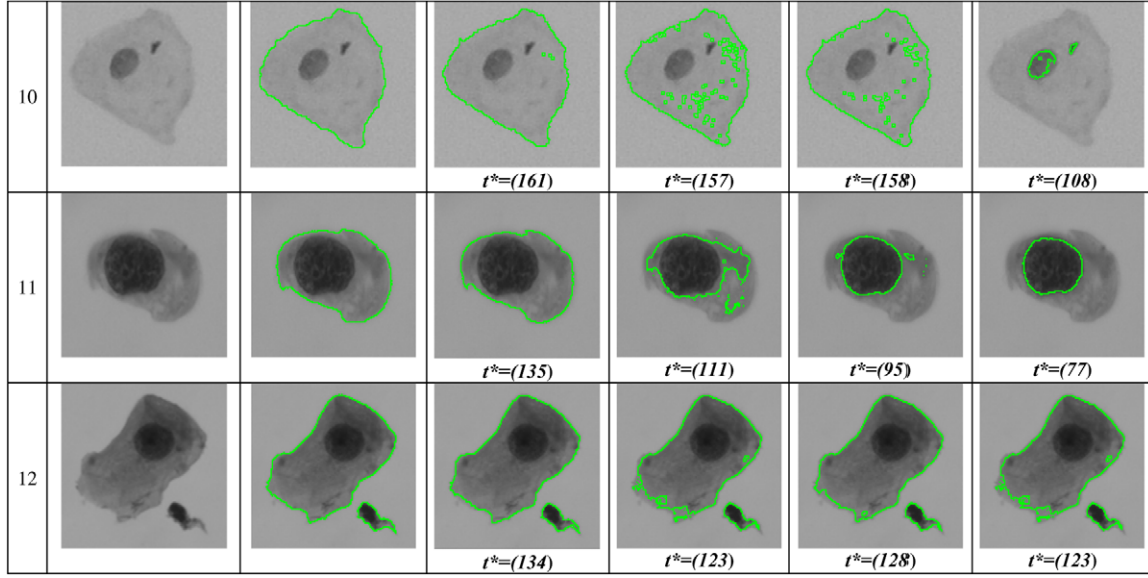


Figure 15. (Continued.)

Table IV. Average ME, RAE, and MHD of the segmentations obtained in Experiment 4.

Measure	TGISM	MWCSD	OTM	VEM	MCVTM
ME	0.018426	0.01941	0.04222	0.03058	0.03735
RAE	0.098582	0.14029	0.16176	0.19491	0.10809
MHD	2.587434	2.58874	4.98858	4.3046	2.75788

where n_1 , n_2 , and n_3 are the numbers of 1's in the binary strings s_1 , s_2 , and s_3 , respectively. gap_1 , gap_2 , and gap_3 are the maximal estimation errors of r_1 , r_2 , and r_3 , respectively. gap_1 , gap_2 , and gap_3 are three given constants.

The GBPD uses the accumulated historical data to train and determine the most appropriate r_1 , r_2 , and r_3 values. For each chromosome, the TGISM is used to segment objects from the accumulated historical images based on the r_1 , r_2 , and r_3 values derived from the chromosome. Then, the average misclassification error (ME) of the segmented objects to the objects manually drawn by human experts is used as the measure of fitness of the chromosome. In the GBPD, N chromosomes are first generated at random. Then, the GBPD evolves the fittest solution by repeatedly executing mutation, crossover, and selection operations, and N chromosomes with the optimal fitness are kept in each run until the fitness among the reserved chromosomes does not vary by a fixed amount from run to run.

In a mutation operation, for each of the N reserved chromosomes, the GBPD randomly specifies three bits b_1 , b_2 , and b_3 , from s_1 , s_2 , and s_3 , respectively, and replaces b_1 , b_2 , and b_3 with $\neg b_1$, $\neg b_2$, and $\neg b_3$, respectively, to generate a new chromosome, where $\neg$ is the "NOT" operator.

In a crossover operation, the GBPD randomly designates N' pairs of chromosomes from the N reserved chromosomes. Let $Ch[i..j]$ be the substring consisting of the i th to j th bits in Ch . Let $|S|$ be the length of string S . For each chromosome pair (Ch_1, Ch_2) , the GBPD concatenates $Ch_1 \left[1.. \left\lfloor \frac{|s_1|}{2} \right\rfloor \right]$,

$Ch_2 \left[\left(\left\lfloor \frac{|s_1|}{2} \right\rfloor + 1 \right) .. |s_1| \right]$, $Ch_1 \left[1.. \left\lfloor \frac{|s_2|}{2} \right\rfloor \right]$, $Ch_2 \left[\left(\left\lfloor \frac{|s_2|}{2} \right\rfloor + 1 \right) .. |s_2| \right]$, $Ch_1 \left[1.. \left\lfloor \frac{|s_3|}{2} \right\rfloor \right]$, and $Ch_2 \left[\left(\left\lfloor \frac{|s_3|}{2} \right\rfloor + 1 \right) .. |s_3| \right]$ into a new chromosome, and $Ch_1 \left[\left(\left\lfloor \frac{|s_1|}{2} \right\rfloor + 1 \right) .. |s_1| \right]$, $Ch_2 \left[1.. \left\lfloor \frac{|s_1|}{2} \right\rfloor \right]$, $Ch_1 \left[\left(\left\lfloor \frac{|s_2|}{2} \right\rfloor + 1 \right) .. |s_2| \right]$, $Ch_2 \left[1.. \left\lfloor \frac{|s_2|}{2} \right\rfloor \right]$, $Ch_1 \left[\left(\left\lfloor \frac{|s_3|}{2} \right\rfloor + 1 \right) .. |s_3| \right]$, and $Ch_2 \left[1.. \left\lfloor \frac{|s_3|}{2} \right\rfloor \right]$ into another new chromosome.

In a selection operation, N optimal chromosomes are selected from the N reserved chromosomes obtained in the previous iteration, the N and $2 \times N'$ chromosomes generated in mutation and crossover operations. The GBPD repeatedly executes the three operations, mutation, crossover, and selection, until the related fitnesses of the reserved N chromosomes are very close. After that, r_1 , r_2 , and r_3 can be calculated by Eq. (21). In the GBPD, the bigger $|s_1|$, $|s_2|$, $|s_3|$ and the smaller gap_1 , gap_2 , gap_3 are, the more suitable values of r_1 , r_2 , and r_3 can be derived. However, the GBPD needs more time to compute the most suitable r_1 , r_2 , and r_3 .

As an example, let $|s_1| = |s_2| = |s_3| = 14$ and $gap_1 = gap_2 = gap_3 = 0.1$. Figure 10(a) shows a representative chromosome Ch . Derived from Ch , $r_1 = 0.1 \times 4 = 0.4$; $r_2 = 0.1 \times 6 = 0.6$; and $r_3 = 0.1 \times 5 = 0.5$. Fig. 10(b) shows a new chromosome created from Ch by the mutation operation, where the bits underlined are the randomly

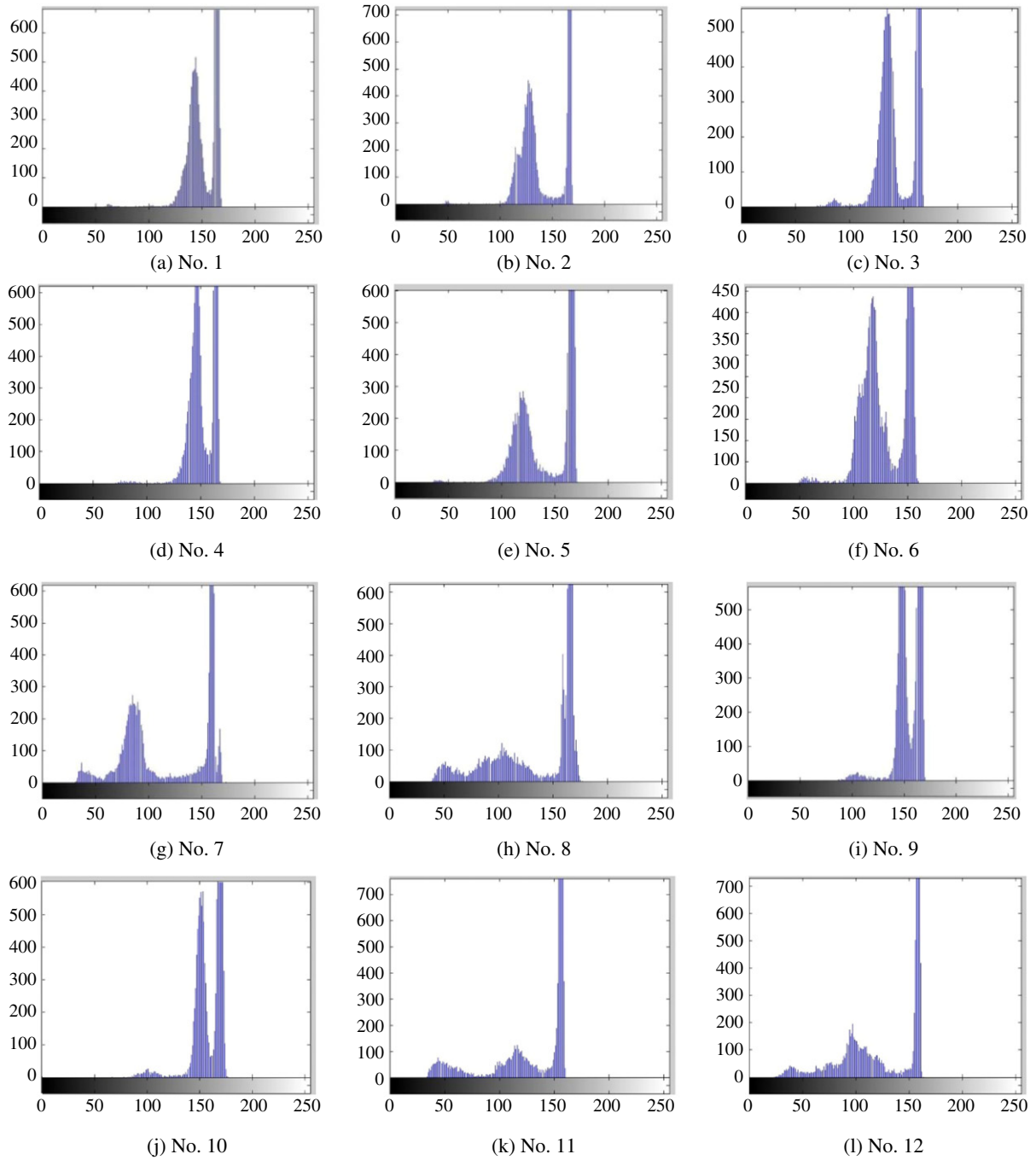


Figure 16. Gray-level histograms of the images in Fig. 15.

Table V. Segmentation error obtained by the TGISM, OTM, VEM, and MCVTM.

Measure	TGISM	OTM	VEM	MCVTM
ME	0.00256	0.00622	0.00335	0.00170
RAE	0.00110	0.01768	0.01193	0.00506
MHD	0.11244	0.15717	0.57551	0.26400

Table VI. Segmentation error obtained by the TGISM, GVF-ACM, and CNC.

Measure	TGISM	GVF-ACM	CNC
ME	0.0184	0.0617	0.0244
RAE	0.0986	0.1743	0.1270
MHD	2.5874	2.8973	2.6701
Average running time (s)	2.48	5.67	5.13

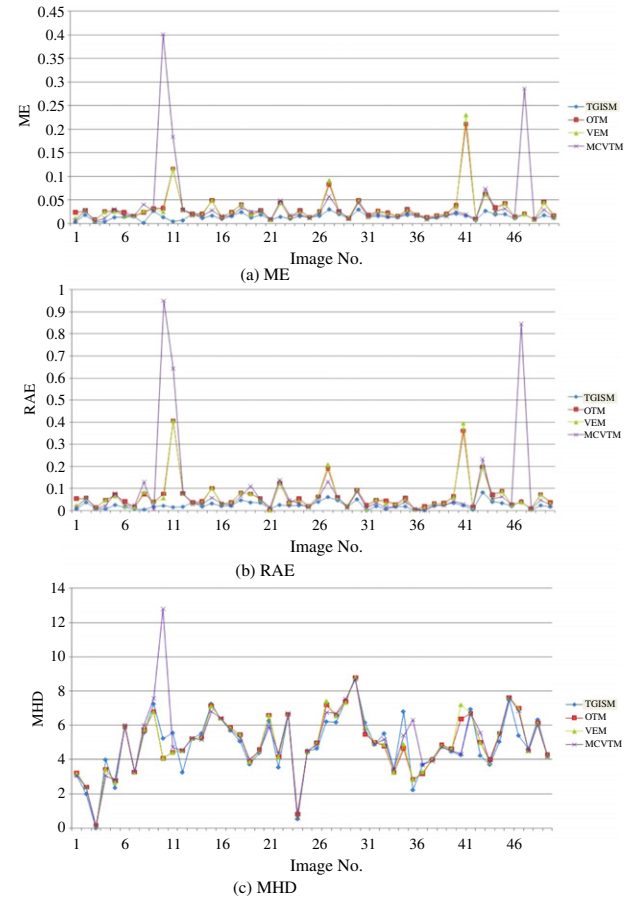
selected bits b_1 , b_2 , and b_3 . Fig. 10(c) displays two new chromosomes, Ch'_1 and Ch'_2 , generated by the crossover operation from the two chromosomes, Ch_1 and Ch_2 .

EXPERIMENTS

Experiments are performed to evaluate the performance of the proposed TGISM as compared with OTM, VEM and MCVTM. In the following experiments, the parameters N , N' , $|s_1|$, $|s_2|$, $|s_3|$, gap_1 , gap_2 , and gap_3 in the GBPD are set to 50, 50, 40, 40, 40, 0.1, 0.1, and 0.1, respectively. m_M in the TGISM is set to 5. The first experiment is to detect the influence of noise on the segmentation methods. This experiment used four 182×182 images with different background textures as test images. Figure 11 displays the four test images. The gray-level histograms of the four test images are shown in Figure 12. In this experiment, Images 1 and 2 in Fig. 11 are used to compute the optimal parameters, $r_1 = 2.2$, $r_2 = 2.0$, and $r_3 = 0$, via the GBPD.

Since the test images in Fig. 11 contain a significant amount of noise with large gradients, the gradient and GVF features are not helpful for determining the appropriate threshold. Hence, the GBPD assigns a value of 0 to r_3 . Fig. 11 shows the segmentation results and the manually drawn target contours. Table I demonstrates the segmentation errors compared to the target contours. It can be seen in Table I that the TGISM produces lower segmentation errors than other segmentation methods. In Fig. 11, since the number of object pixels is much less than the number of background pixels in the test images, OTM cannot produce good segmentation results.

Experiment 2 is designed to explore the performance of TGISM in determining a suitable threshold for separating objects from an image in which the class intervals of the gray-levels in objects and image background are similar. The three images in Figure 13 were used as the test images, each consisting of 128×128 pixels. Figure 14 shows their gray-level histograms. In this experiment, the first image is used to train the parameters $r_1 = 1.6$, $r_2 = 0$, and $r_3 = 0.3$ by GBPD. Since the class intervals of the two classes in Fig. 14(a) are quite close to each other, it is ineffective in deciding the threshold. Therefore, in GBPD, r_2 is set to 0. Fig. 14 shows the segmentation results, and Table II summarizes the ME, RAE, and MHD values obtained in this experiment compared to the target contours. Fig. 14(c) shows the gray-level histogram of the image in Fig. 13. Since, the variances of the gray-levels in the objects and in the background of Image 3 in Fig. 13

**Figure 17.** ME, RAE, and MHD values obtained in Experiment 3.

are very different, the MCVTM cannot produce an effective segmentation.

The third experiment used 50 128×128 pixels cervical smear images as test images. Figure 15 shows the first 12 images. Each of the cervical smear images contains one cell, consisting of a cytoplasm and a nucleus, as well as the background. Therefore, each cervical smear image generally has more than one peak in its gray-level histogram. Some users may desire to segment only the cells from the background. Others may prefer to segment off both the cytoplasts and the nuclei.

This experiment was to segment the cells from the background. Images 1 through 12 in Fig. 15 were used to compute the most suitable parameters, $r_1 = 1.5$, $r_2 = 0.3$, and $r_3 = 0.2$, via the GBPD. Figs. 15 and 16 illustrate the segmentation results and the gray-level histograms of the test

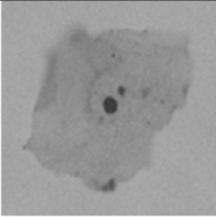
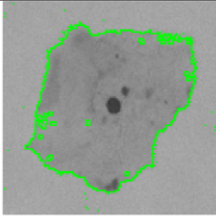
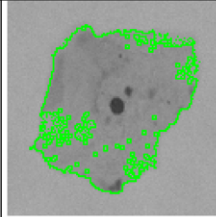
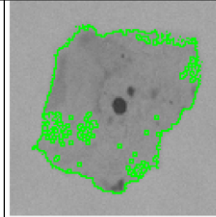
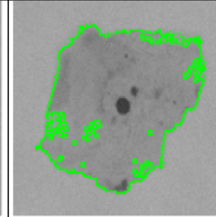
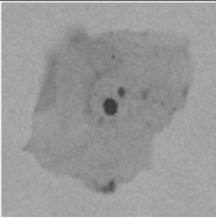
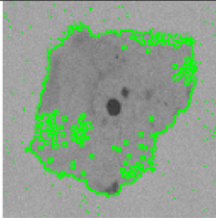
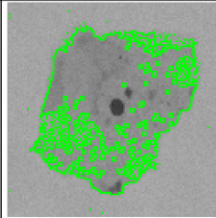
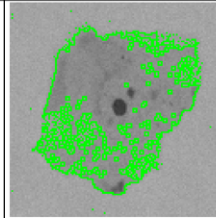
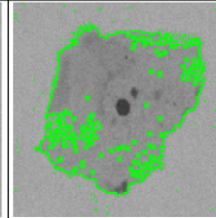
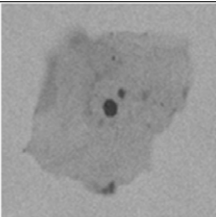
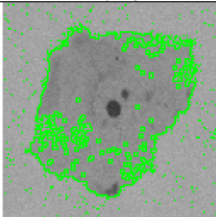
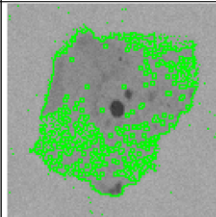
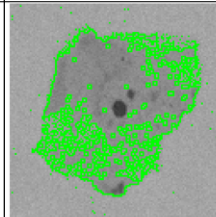
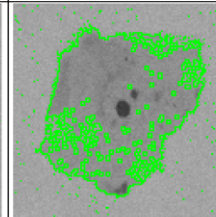
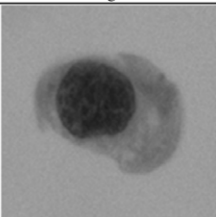
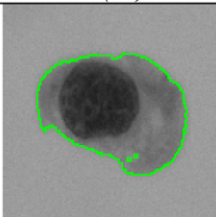
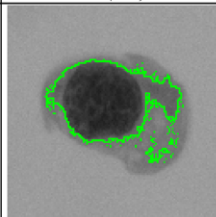
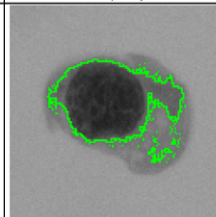
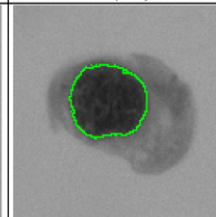
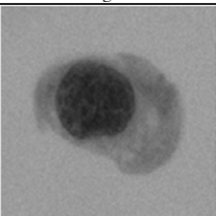
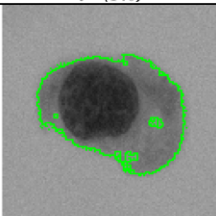
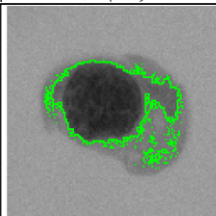
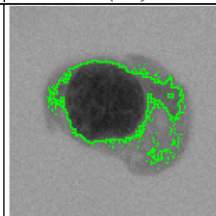
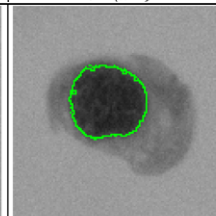
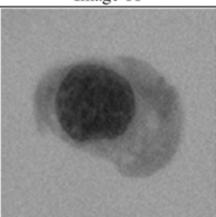
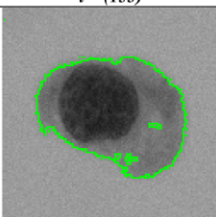
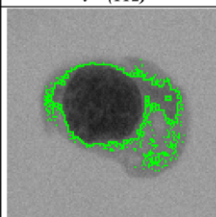
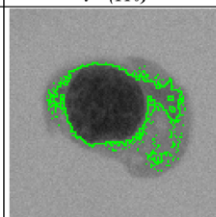
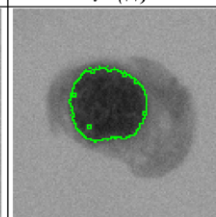
GN	Image	TGISM	OTM	VEM	MCVTM
1%					
	Image 1	$t^*=(155)$	$t^*=(151)$	$t^*=(152)$	$t^*=(153)$
1.5%					
	Image 1	$t^*=(154)$	$t^*=(151)$	$t^*=(151)$	$t^*=(153)$
2%					
	Image 1	$t^*=(154)$	$t^*=(151)$	$t^*=(151)$	$t^*=(153)$
1%					
	Image 11	$t^*=(140)$	$t^*=(111)$	$t^*=(111)$	$t^*=(177)$
1.5%					
	Image 11	$t^*=(133)$	$t^*=(112)$	$t^*=(110)$	$t^*=(77)$
2%					
	Image 11	$t^*=(138)$	$t^*=(112)$	$t^*=(110)$	$t^*=(177)$

Figure 18. The two cervical smear images obtained by adding Gaussian noise and the contours generated by the TGISM, OTM, VEM, and MCVTM.

images, respectively. Figure 17 and Table III illustrate the errors of segmenting objects from the 50 test images.

The fourth experiment is designed to investigate the robustness of the TGISM ($r_1 = 1.5$, $r_2 = 0.3$, and $r_3 = 0.2$) in resisting noise variation. The task is to split the cytoplasm out from the two 128×128 cervical smear images in Fig. 15, in which 1%, 1.5%, and 2% Gaussian noise is added, as shown in Figure 18. Figure 19 shows the corresponding

gray-level histograms with added noise. Figs. 18 and 20 illustrate the segmentation results, peak signal-to-noise ratio (PSNR)¹⁷ relative to the original images, and segmentation errors, respectively. The experimental results demonstrate that the TGISM provides better performance than OTM, VEM, and MCVTM in resisting noise variation.

The next task is to segment out the cytoplasm and nucleus concurrently from a cervical smear cell image. In

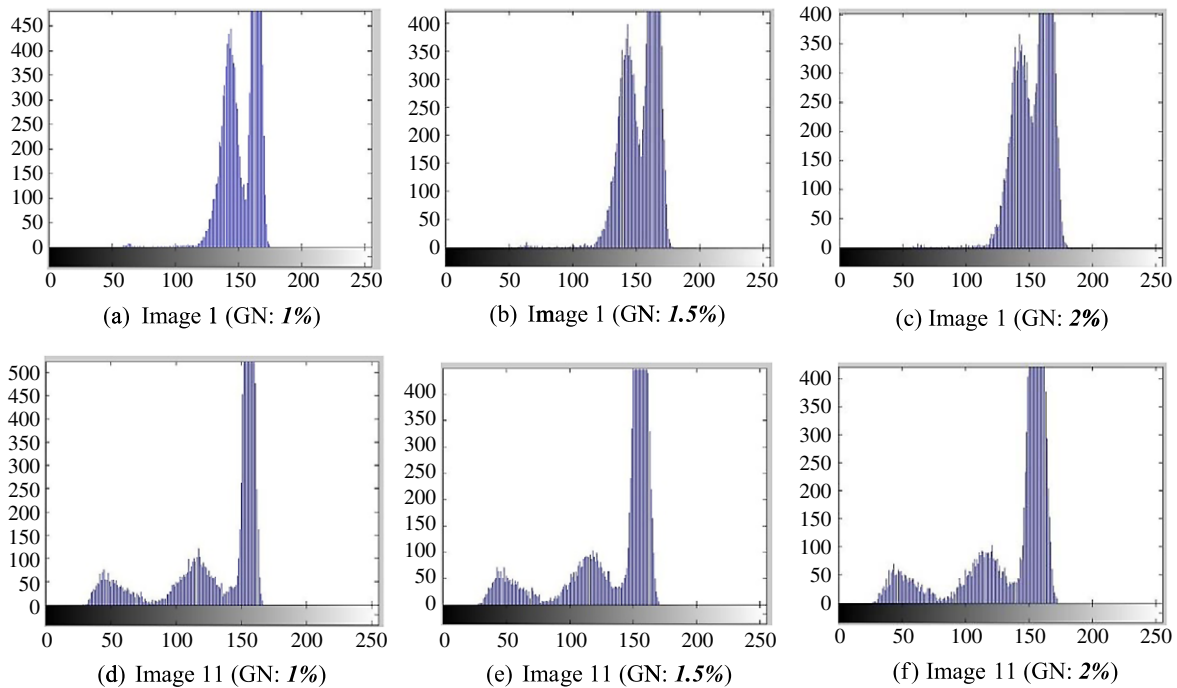


Figure 19. Gray-level histograms of the images in Fig. 20.

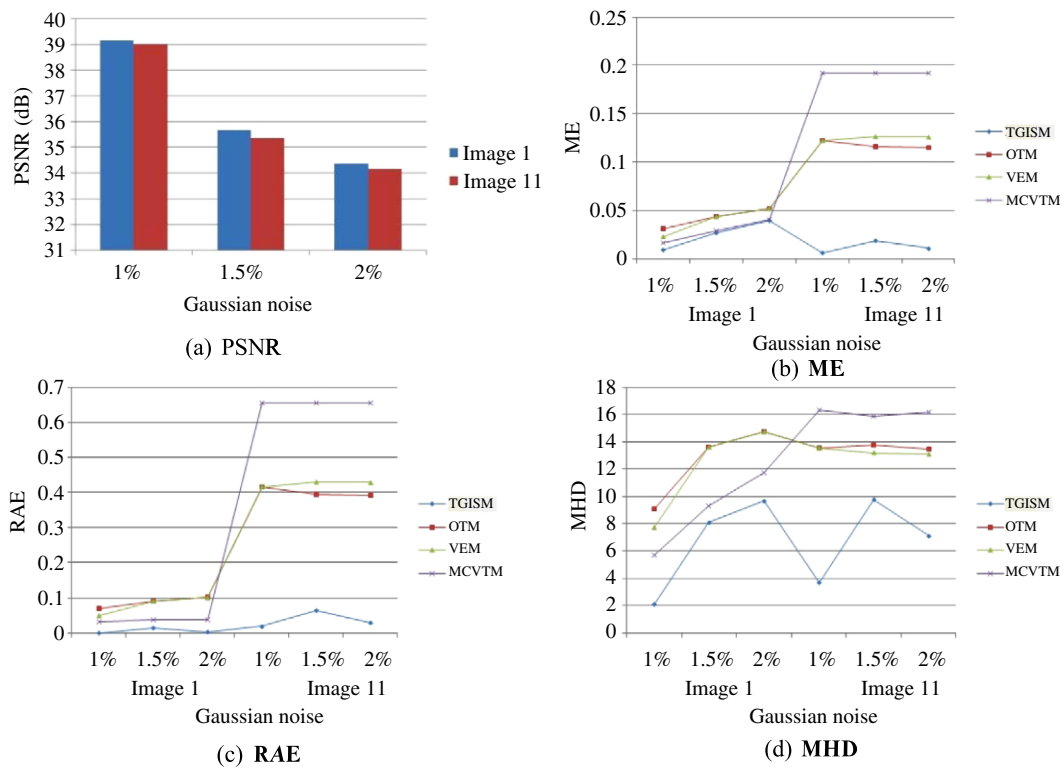


Figure 20. PSNR and segmentation errors obtained in Experiment 4.

this task, it is necessary to determine two thresholds. In this experiment, 50 cervical smear images are used as test images. Images 1 through 12 are used to compute the most suitable $r_1 = 2.6$, $r_2 = 1.1$, and $r_3 = 0.8$ by the GBPD. Figure 21

displays the objects segmented from the 12 cervical smear images in Fig. 15. Figure 22 and Table IV state the average segmentation errors for extracting objects from the 50 test images. Since the user can adjust the values of r_1 , r_2 , and r_3

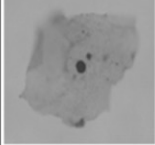
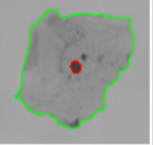
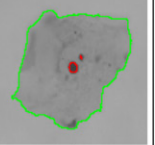
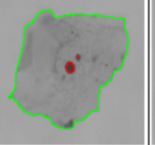
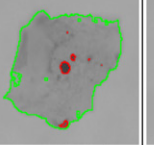
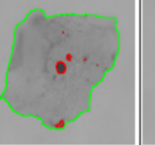
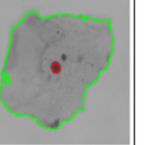
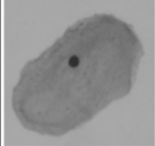
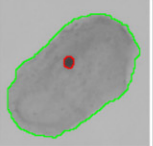
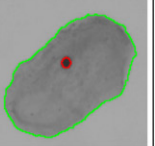
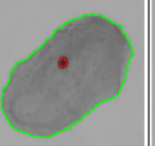
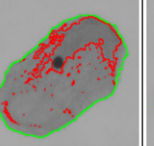
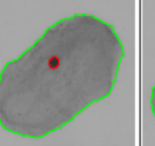
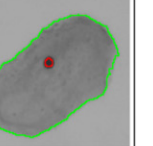
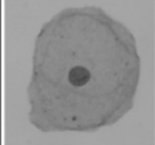
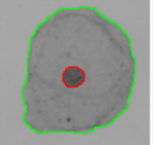
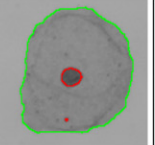
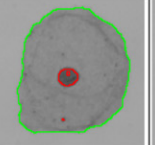
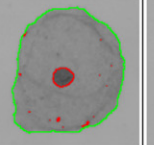
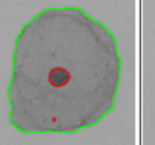
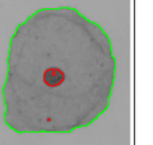
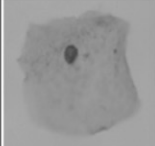
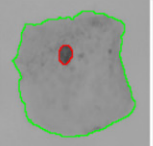
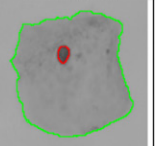
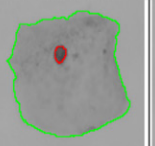
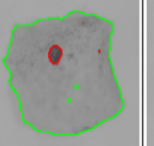
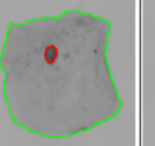
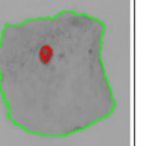
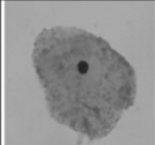
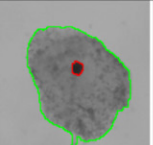
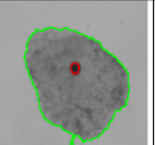
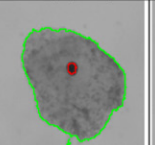
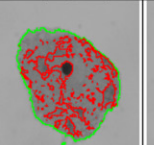
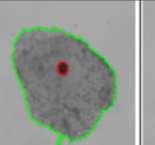
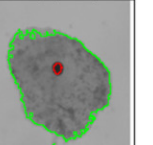
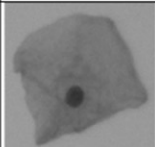
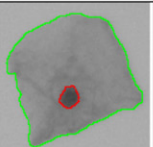
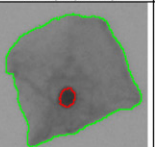
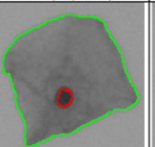
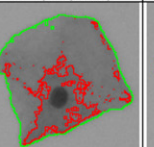
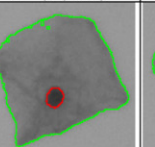
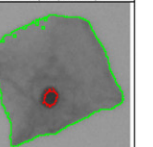
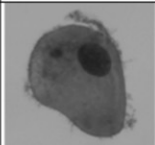
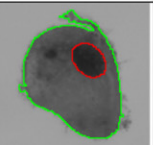
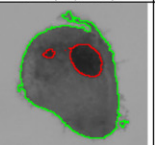
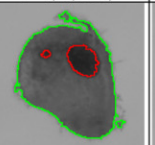
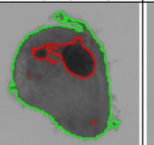
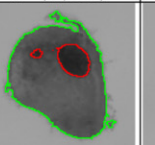
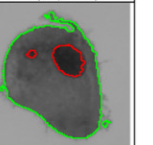
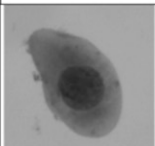
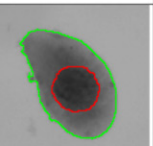
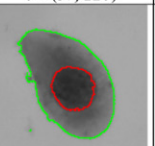
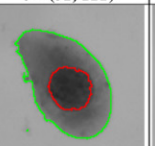
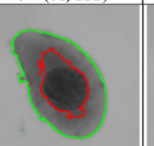
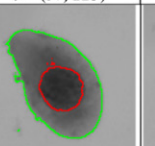
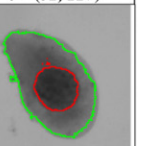
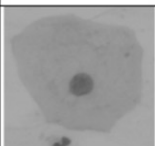
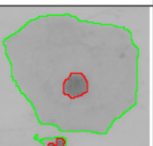
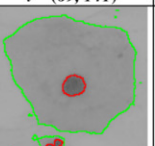
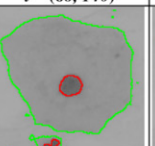
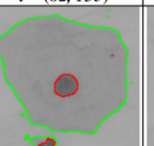
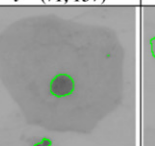
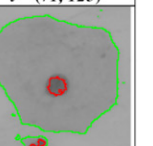
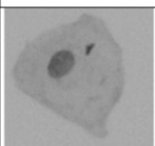
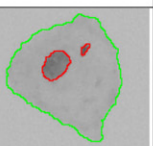
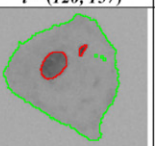
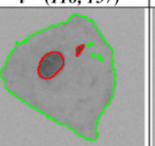
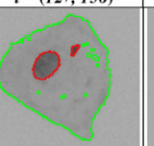
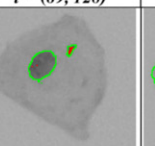
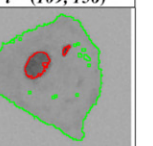
No.	Original image	Target Contour	TGISM	MWCSD	OTM	VEM	MCVTM
1			 $t^*=(78, 155)$	 $t^*=(72, 156)$	 $t^*=(112, 152)$	 $t^*=(112, 158)$	 $t^*=(69, 153)$
2			 $t^*=(70, 151)$	 $t^*=(55, 152)$	 $t^*=(121, 147)$	 $t^*=(91, 151)$	 $t^*=(55, 145)$
3			 $t^*=(99, 150)$	 $t^*=(92, 150)$	 $t^*=(111, 148)$	 $t^*=(100, 148)$	 $t^*=(92, 149)$
4			 $t^*=(98, 157)$	 $t^*=(95, 157)$	 $t^*=(115, 153)$	 $t^*=(106, 158)$	 $t^*=(85, 155)$
5			 $t^*=(50, 151)$	 $t^*=(50, 147)$	 $t^*=(114, 144)$	 $t^*=(79, 150)$	 $t^*=(48, 138)$
6			 $t^*=(70, 139)$	 $t^*=(66, 136)$	 $t^*=(109, 135)$	 $t^*=(79, 135)$	 $t^*=(63, 134)$
7			 $t^*=(57, 120)$	 $t^*=(51, 121)$	 $t^*=(68, 122)$	 $t^*=(57, 123)$	 $t^*=(52, 127)$
8			 $t^*=(69, 141)$	 $t^*=(68, 140)$	 $t^*=(82, 135)$	 $t^*=(71, 137)$	 $t^*=(71, 125)$
9			 $t^*=(120, 157)$	 $t^*=(118, 157)$	 $t^*=(127, 156)$	 $t^*=(89, 128)$	 $t^*=(109, 156)$
10			 $t^*=(121, 161)$	 $t^*=(117, 160)$	 $t^*=(126, 159)$	 $t^*=(85, 133)$	 $t^*=(111, 159)$

Figure 21. Twelve cervical smear images, the target contours, and the contours cut by the TGISM, MWCSD, OTM, VEM, and MCVTM.

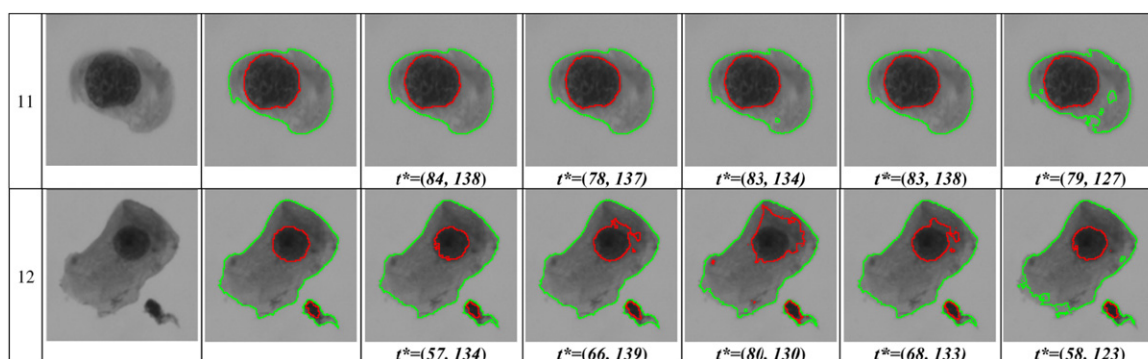


Figure 21. (Continued.)

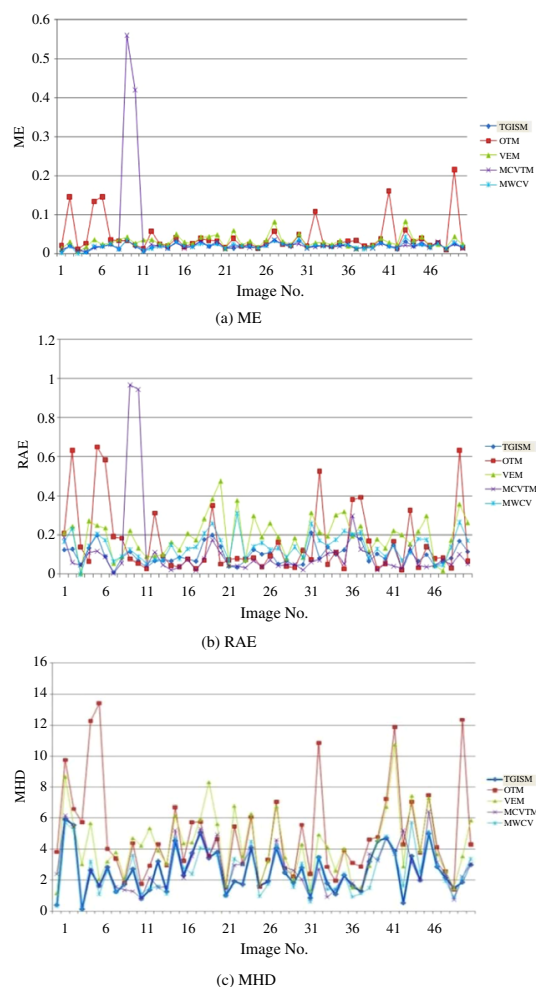


Figure 22. ME, RAE, and MHD values of segmentations obtained in Experiment 4.

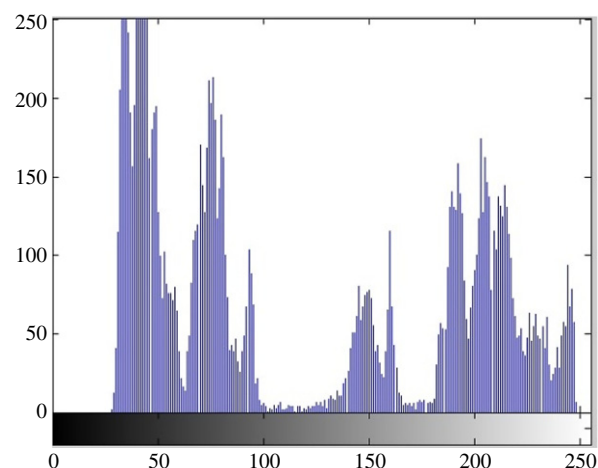


Figure 24. Gray-level histogram of the original image in Fig. 25.

to satisfy her or his requirements via the GBPD, the TGISM can provide more flexible segmentation results than OTM, VEM and MCVTM.

The sixth experiment is conducted to investigate the influence of minimizing within-class standard deviation (MWCS) on image segmentation. This experiment was to simultaneously detect the cytoplasm and nuclear contours on the 50 cervical smear images. In this experiment, r_3 is set to 0 and Images 1 through 12 in Fig. 17 were used to obtain the most appropriate $r_1 = 2.9$ and $r_2 = 1.7$ via the GBPD. Fig. 21 demonstrates the segmented objects from the first 12 images. Fig. 22 and Table IV illustrate the segmentation errors obtained in this experiment by segmenting objects from the 50 cervical smear images. Overall, the experimental results demonstrate that the TGISM is superior to MWCS

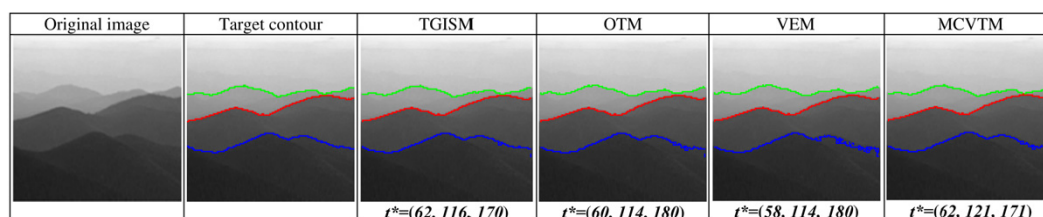


Figure 23. Original image, target contour, and detected contours.

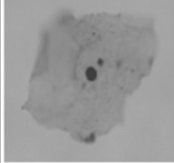
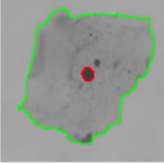
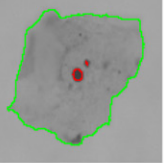
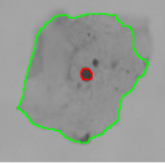
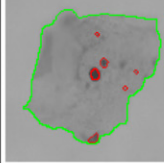
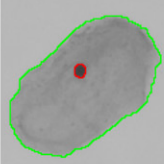
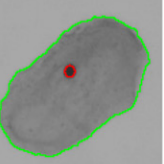
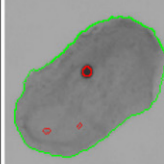
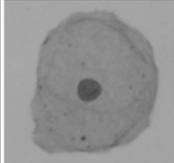
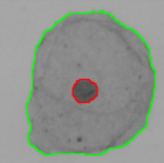
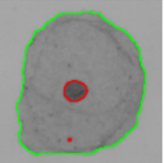
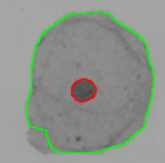
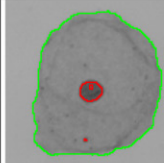
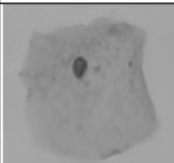
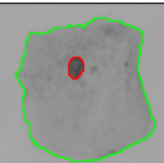
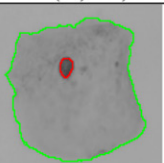
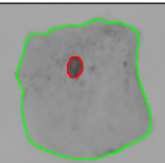
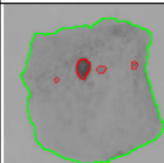
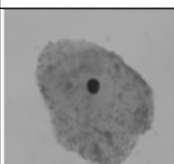
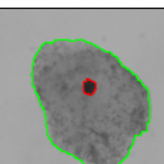
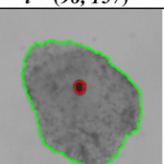
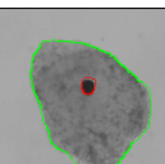
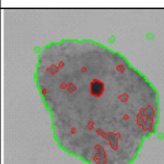
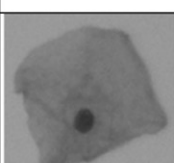
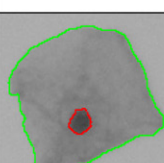
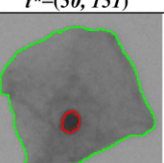
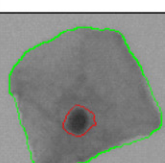
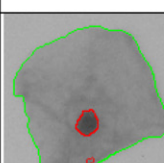
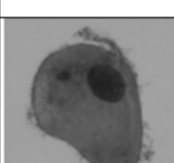
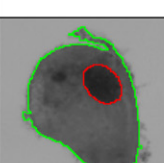
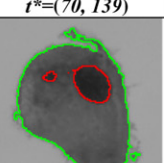
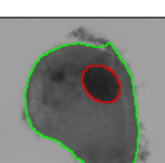
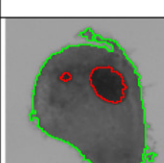
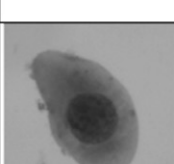
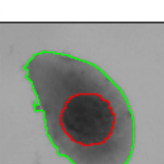
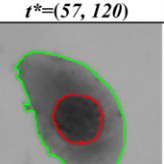
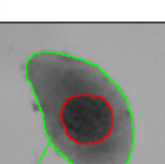
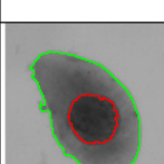
No.	Original image	Target Contour	TGISM	GVF-ACM	NCC
1			 $t^*=(78, 155)$		
2			 $t^*=(70, 151)$		
3			 $t^*=(99, 150)$		
4			 $t^*=(98, 157)$		
5			 $t^*=(50, 151)$		
6			 $t^*=(70, 139)$		
7			 $t^*=(57, 120)$		
8			 $t^*=(69, 141)$		

Figure 25. Twelve cervical smear images, the target contours, and the contour cut by the TGISM, GVF-ACM, and CNC.

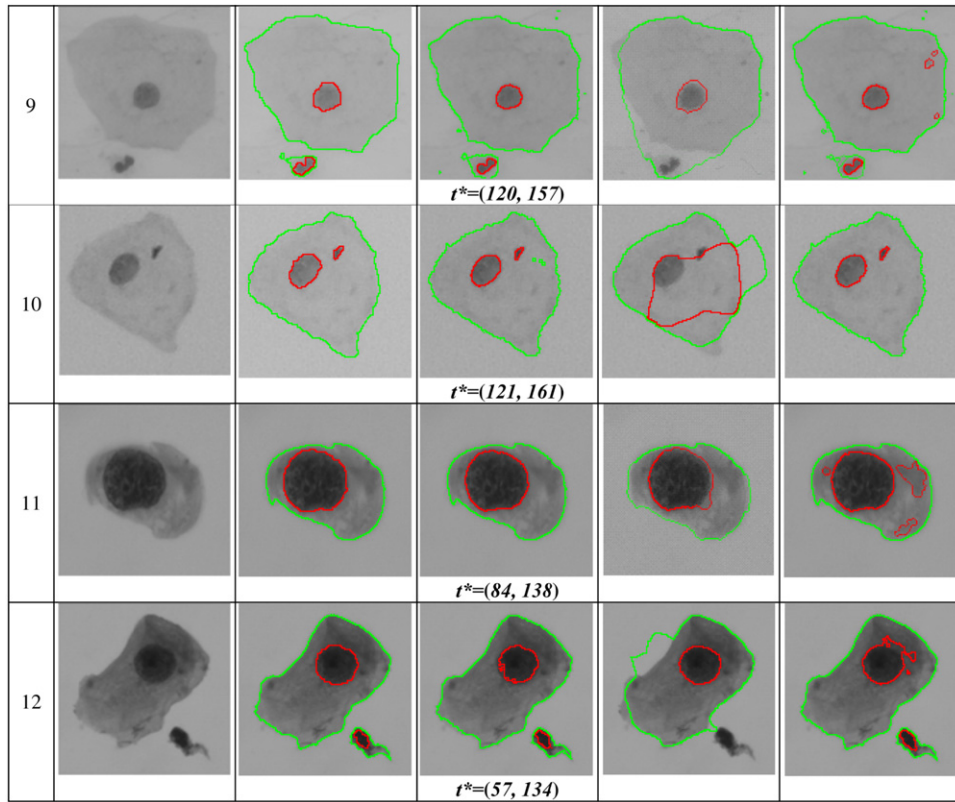


Figure 25. (Continued.)

and that MWCSO is better than OTM, VEM, and MCVTM in image segmentation.

The seventh experiment was to scrutinize the performance of the TGISM in providing multiple thresholds. In this experiment, the optimal parameters r_1 , r_2 , and r_3 were set to 2.8, 1.6, and 2.7, respectively. Figures 23 and 24 show the test image and its gray-level histogram, respectively. Fig. 23 and Table V display the experimental results.

Tsai et al.¹³ provided a cytoplasm and nucleus contour (CNC) detector to sever the nucleus and cytoplasm from a cervical smear image. The last experiment is to compare the performances of the TGISM, GVF-ACM, and CNC in segmenting cytoplasm and nucleus from a cervical smear cell image. In this experiment, the cervical smear images used in the fourth experiment are used as test images. Figure 25 and Table VI show the experimental results. The average running time for extracting the object from an image is summarized in Table VI. It can be seen that the TGISM is more efficient and effective than the GVF-ACM¹¹ and CNC¹³ in object segmentation.

CONCLUSIONS

Thresholding is one of the most commonly used techniques in extracting objects from an image. In this article, the MWCSO method is proposed to determine the appropriate threshold for effective and efficient segmentation. The TGISM, which takes into account the minimizing within-class standard deviation, gradient of object contour,

and GVF features, is proposed to precisely extract objects from an image. The GBPD is shown to generate the appropriate parameters used in the TGISM. Experimental results confirmed that the TGISM does provide better segmentation results than OTM, VEM, and MCVTM.

REFERENCES

- 1 Z. A. Agus and A. Akira, "Image segmentation by histogram thresholding using hierarchical cluster analysis," *Pattern Recognit. Lett.* **27**, 1515 (2006).
- 2 C. A. Glasbey, "An analysis of histogram-based thresholding algorithms," *CVGIP: Graph. Models Image Process.* **55**, 532 (1993).
- 3 S. U. Lee, S. Y. Chung, and R. H. Park, "A comparative performance study of several global thresholding techniques for segmentation," *Comput. Vision, Graph. Image Process.* **52**, 171 (1993).
- 4 P. K. Sahoo, S. Soltani, A. K. Wong, and Y. C. Chan, "A survey of thresholding techniques," *Comput. Vision, Graph. Image Process.* **41**, 233 (1988).
- 5 M. Sezgin and B. Sanlur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imaging* **13**, 146 (2004).
- 6 N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.* **9**, 62 (1979).
- 7 P. S. Liao, T. S. Chen, and P. C. Chung, "A fast algorithm for multilevel thresholding," *J. Inf. Sci. Eng.* **17**, 713 (2001).
- 8 Z. Hou, Q. Hu, and W. L. Nowinski, "On minimum variance thresholding," *Pattern Recognit. Lett.* **27**, 1732 (2006).
- 9 H. D. Cheng and Y. Chen, "Fuzzy partition of two-dimensional histogram and its application to thresholding," *Pattern Recognit.* **32**, 825 (1999).
- 10 J. S. Lee and M. C. K. Yang, "Threshold selection using estimates from truncated normal distribution," *IEEE Trans. Syst. Man Cybern.* **19**, 422 (1989).
- 11 H. F. Ng, "Automatic thresholding for defect detection," *Pattern Recognit. Lett.* **27**, 1644 (2006).

- ¹² R. J. Qian and T. S. Huang, "Optimal edge detection in two dimensional images," *IEEE Trans. Image Process.* **5**, 1215 (1996).
- ¹³ M. H. Tsai, Y. K. Chan, Z. Z. Lin, S. F. Yang-Mao, and P. C. Huang, "Nucleus and cytoplasm contour detector of cervical smear image," *Pattern Recognit. Lett.* **29**, 1441 (2008).
- ¹⁴ C. Xu and L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Trans. Image Process.* **7**, 359 (1998).
- ¹⁵ K. F. Man, K. S. Tang, and S. Kwong, *Genetic Algorithms: Concepts and Designs* (Springer-Verlag, New York, 1999).
- ¹⁶ X. Wu, H. Gao, H. Heo, O. Chae, J. Cho, S. Lee, and Y. K. Lee, "Improved B-spline contour fitting using genetic algorithm for the segmentation of dental computerized tomography image sequences," *J. Imaging Sci. Technol.* **51**, 328 (2007).
- ¹⁷ Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognit.* **29**, 1335 (1996).