Journal of Electronic Imaging 13(3), 602–612 (July 2004).

Robust region segmentation based on a pseudo-distance map

Byoung-Ki Jeon
Jeong-Hun Jang
Ki-Sang Hong
Postech
Division of Electrical and Computer Engineering
San 31 Hyoja-Dong, Pohang
Korea, 790-784
E-mail: {standard,jeonghun,hongks}@postech.ac.kr

Abstract. *We present a robust region segmentation method based on a pseudo-distance map (PDM) that uses a watershed algorithm as a segmentation tool. The PDM is a regularized version of a Euclidean distance map (EDM) directly computed from the edge-strength function (ESF) of an input image without edge detection, which involves a thresholding operation. This unavoidably causes useful region boundary information loss from the original image. We show that applying the watershed algorithm to the PDM significantly reduces oversegmentation, and the final segmentation results obtained by a simple region-merging process are more accurate and meaningful and less sensitive to noise than those of the gradient-based or EDM-based methods. We also propose a simple and efficient region-merging criterion that considers both boundary strengths and inner intensities of regions to be merged. We tested and verified the robustness of our method with a variety of synthetic and real images.* © *2004 SPIE and IS&T.* [DOI: 10.1117/1.1758952]

1 Introduction

A Euclidean distance map (EDM) is an image where each point is assigned a Euclidean distance to the nearest shape boundary point. Instead of an original image, the EDM obtained is commonly used in vision applications, such as skeletonization and segmentation. For example, since Blum's proposal,[1] a medial axis is generally accepted as a definition of a skeleton in the literature, and as a skeletonization method, the medial axis is extracted[2,3] by detecting ridges of the EDM. In the applications in Refs. 4 and 5, for region segmentation the watershed transformation was applied to the EDM constructed from an edge image, since it has the characteristic of detecting meaningful boundaries by separating connected or overlapping blobs when applied to the corresponding EDM, as shown in Fig. 1.

The common requirement of the preceding approaches is that the boundary contour of a shape must be determined before its skeletonization or segmentation. In the case of a binary image, the requirement is trivial because the boundary contour of the binary image is always clearly determined by its definition. However, this becomes an obstacle when attempting to skeletonize or segment a gray-scale image with the methods using the EDM, since no clear boundary of a shape or a region is defined in gray-scale images. If one tries to skeletonize or segment gray-scale images from the EDM, the boundaries of the regions should be extracted directly with edge detection techniques. However, since edge detection always involves a thresholding operation, it is unavoidable that some useful information will be lost from the original image, which seems to be necessary to extract a reliable skeleton or to accurately segment regions. Some spurious edges resulting from the thresholding operation may change the shapes of skeletons and cause noisy segmented regions. Moreover, these methods are usually sensitive to a change in the thresholding values of the edge detection process.

In previous work,[6] we proposed a new tool, called a pseudo-distance map (PDM), to directly extract skeletons from gray-scale images without region segmentation or edge detection. In this paper, we present a robust region segmentation method based on the PDM using a watershed algorithm as a segmentation technique and show that our method provides segmentation results superior to conventional methods. A typical approach for segmenting a gray-scale image with watershed transformation is to make use of its gradient image as an input to the transformation, since high gradients constitute watershed lines that correspond to the region boundaries of the gray-scale image. In our proposed method, however, we utilize the PDM as an input to the watershed algorithm. This maintains the advantage of the EDM, which can segment meaningful boundaries, as depicted in Fig. 1, while overcoming the shortcomings of the EDM caused by the thresholding operation.

A PDM can be regarded as a relaxed and regularized version of an EDM. While the EDM is obtained from an edge image, the PDM is directly computed from the edge-strength function (ESF) of a given gray-scale image without thresholding it. An ESF is a smooth function that approaches the value of one at the shape boundary and decays rapidly to zero while receding from the boundary. The value of the PDM is almost equal to zero where edge

Paper 03102 received Jul. 16, 2003; revised manuscript received Dec. 2, 2003; accepted for publication Dec. 22, 2003.
1017-9909/2004/$15.00 © 2004 SPIE and IS&T.

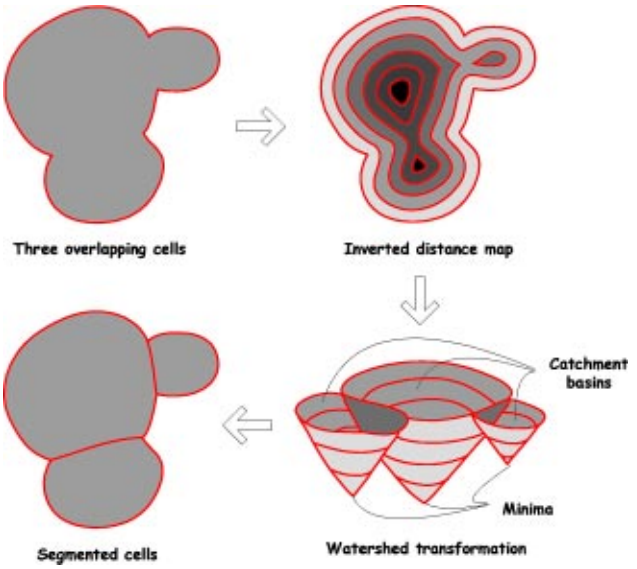


Fig. 1 Segmentation of overlapping cells by applying the watershed transformation to the inverted distance map.[5]

strength is relatively large, and the function has nearly constant slopes at the points with small edge strength, except at the positions where two opposite slopes meet, which actually correspond to the skeleton of the input image. Given an ESF, the PDM is obtained by numerically solving the partial differential equations (PDEs) that are derived from the energy functional we propose. Our method completely avoids performing the edge detection process by replacing the edge map and the EDM with the ESF and the PDM, respectively. Therefore, the proposed PDM-based method fully utilizes the edge strength information of the given image and contains the regularization effect by the variational formulation. Hence, the application of watershed transformation to the inverted PDM significantly reduces the oversegmentation problem caused by noisy gradients or spurious edges, and usually produces more accurate and less noisy initial segmentation results than the gradient- or EDM-based methods.

In addition, we propose a simple and efficient region-merging criterion that considers both boundary strengths and inner intensities of the regions to be merged. Through several experiments, we show that our region-merging method, along with our initial segmentation method, gives robust and reliable final segmentation results.

This paper is organized as follows. Section 2, describes how to obtain the ESF and the PDM, including some ideas for faster computation of the PDM. In Sec. 3, the overall procedure of the proposed method is provided, including our new region-merging criterion. Experimental results are shown in Sec. 4, where we compare the performance of our method with those of the conventional gradient- or EDM-based methods and show the robustness of the PDM-based method. Finally, conclusions and discussions are given in Sec. 5.

2 ESF and PDM

2.1 *EDM versus PDM*

An EDM is defined as an image where each pixel is assigned the value of the distance to the nearest object boundary pixel. We developed a PDM as a regularized version of the conventional EDM.[6]

Note the connection between the EDM and the eikonal equation. Consider a curve evolving with the PDE

$$\frac{\partial C}{\partial t} = F(x,y)N, \tag{1}$$

where C is a parameterized representation of the curve, N is the unit inward (or outward) normal to the curve, and $F(F>0)$ is a speed function which depends only on the position (x,y). If $T(x,y)$ represents the time at which the curve crosses a point (x,y), it can be shown that $T(x,y)$ satisfies the eikonal equation[7,8]

$$\|\nabla T\|F = 1, \quad T = 0 \quad \text{on} \quad \Gamma, \tag{2}$$

where Γ is the initial location of the curve. The connection between the EDM and the eikonal equation is the fact that the EDM can be interpreted as the solution T of Eq. (2), with F equal to 1 if Γ corresponds to the object boundary. There are several methods for solving Eq. (2), which include, for example, Sethian's fast marching method.[7] However, if the boundary location is not precisely given, and we know only the strength of the boundary at each pixel location, can we compute something similar to T in such a restricted situation? The development of a PDM is motivated by this question.

2.2 *ESF*

As mentioned in Sec. 1, before computing the PDM, we must first obtain the ESF from a given gray-scale image. In this subsection, we introduce a variety of ESFs available. The simplest form of a ESF, v, would be

$$v = g(\|\nabla G_\sigma * I_0\|), \tag{3}$$

where G_σ is a Gaussian kernel of size σ, I_0 is an input gray-scale image, and g is a normalizing function that increases monotonically and maps $[0,\infty]$ to $[0,1]$. The parameter σ controls the overall scale of the ESF. It can be shown that convolving an image with a Gaussian kernel is equivalent to performing isotropic linear diffusion with an initial state set to I_0. Perona and Malik proposed an anisotropic diffusion equation for edge-preserving smoothing:[9]

$$I_t = \nabla \cdot c(\|\nabla I\|)\nabla I \quad \text{with} \quad I = I_0 \quad \text{at} \quad t = 0, \tag{4}$$

where c is called a conductance function. They suggested two conductance functions:

$$c(x,y,t) = \exp\left[-\frac{\|\nabla I(x,y,t)\|^2}{K^2}\right] \quad \text{and} \tag{5}$$

$$c(x,y,t)=\frac{1}{1+\|\nabla I(x,y,t)\|^2/K^2}. \tag{6}$$

In short, K plays the role of a threshold for gradients, that is, if the gradient at (x,y,t) is larger than the predefined value of K then the conductance $c(x,y,t)$ decays rapidly, the image value $I(x,y,t)$ is less smoothed, and its gradient information is preserved. Therefore, the larger K we use, the more gradient information of the given image we lose. The ESF of $I(x,y,t)$ can be computed by negating the role of the conductance function and given by

$$v=1-c(x,y,t). \tag{7}$$

Other anisotropic diffusion equations can be found in Refs. 10 and 11.

Ambrosio and Tortorelli[12] proposed an ESF that is obtained by minimizing the functional

$$E(I,v)=\int \mu(1-v)^2\|\nabla I\|^2+\eta(I-I_0)^2+\frac{\sigma}{2}\|\nabla v\|^2 + \frac{v^2}{2\sigma}\,\mathrm{d}x\,\mathrm{d}y, \tag{8}$$

where I_0 is an input image, I is its smoothed image, and v is an ESF. The two functions I and v that minimize Eq. (8) can be computed using the variational approach,[13] which gives rise to the following coupled PDEs:

$$I_t=\nabla\cdot(1-v)^2\nabla I-\frac{\eta}{\mu}(I-I_0), \tag{9}$$

$$v_t=\nabla^2 v-\frac{v}{\sigma^2}+\frac{2\mu}{\sigma}(1-v)\|\nabla I\|^2. \tag{10}$$

Note that the larger the μ/η, the more the details of the ESF are smoothed. For experiments, we adopted this method, which is more sophisticated than the first two methods based on a Gaussian kernel and a conductance function of an anisotropic diffusion at the cost of additional computational burden. Of course, readers can adopt any other methods[14,15] for obtaining the ESFs.

2.3 *PDM*

In this subsection, we introduce how to compute a PDM from a given ESF.

2.3.1 *One-dimensional formulation*

We begin by introducing an energy functional that will be minimized to obtain a PDM in one dimension. The functional is extended to the 2-D case in the next subsection. Assuming that an ESF $v(x)$ is given, which ranges from 0 to 1 and monotonically increases as an edge at x becomes stronger, the functional is given by

$$E(f)=\int \underbrace{\alpha v[f^2+f_x^2+(b-f_{xx})^2]}_{(A)}+\underbrace{\beta(a^2-f_x^2)^2}_{(B)}\,\mathrm{d}x, \tag{11}$$

where α, β, a, and b are positive constants, and f is the PDM to be computed. The functional works as follows. If v is small, the minimization of E will be dominated by the term (B). This means that the slope of f will approach a or $-a$. If v is large, the term (A) will also affect the minimization of E and the constraints on f will work; that is, the magnitudes of f and f_x should become zero and f_{xx} should approach the positive constant b. These constraints give f a local minima of nearly zero at positions where v is large. Therefore, the overall shape of f will become similar to that of an EDM after the minimization of E.

The function f that minimizes the functional E can be computed by the variational method if the initial form of f is appropriately given. In the variational method, PDEs corresponding to a given energy functional are obtained using the calculus of variations,[13] and the PDEs are discretized to compute their numerical solutions. Note that f_{xx} in Eq. (11) causes the fourth-order derivative of f in the corresponding PDE, which tends to make the PDE noise sensitive. To avoid the use of f_{xx}, we introduce a new function g, which approximates[16] f_x, and incorporate it with Eq. (11):

$$E(f,g)=\int \alpha v[f^2+f_x^2+(b-g_x)^2]+\beta(a^2-f_x^2)^2 + (g-f_x)^2\,\mathrm{d}x. \tag{12}$$

Equation (12) gives rise to two PDEs:

$$f_t=\alpha[(vf_x)_x-vf]+2\beta f_{xx}(3f_x^2-a^2)+(f_{xx}-g_x), \tag{13}$$

$$g_t=\alpha[(vg_x)_x-bv_x]+(f_x-g). \tag{14}$$

We implemented the preceding PDEs using central finite-difference approximations shown in Sec. 2.3.3 and solved them with the initial value of f set to $1-v$. Unfortunately, according to our experiments, the convergence of the PDE in Eq. (13) was inadequate; that is, it was very slow and often stuck to unwanted local extrema. To speed up the convergence and avoid falling into the local extrema, we added a new term to Eq. (13):

$$f_t=\alpha[(vf_x)_x-vf]+2\beta f_{xx}(3f_x^2-a^2)+(f_{xx}-g_x) + \gamma\phi f_{xx}, \tag{15}$$

where γ is a positive constant and

$$\phi=\begin{cases}1 & \text{if } \{f<0\} \quad \text{or } \{f_x\approx 0, f_{xx}>0, v\approx 0\},\\ 0 & \text{otherwise.}\end{cases} \tag{16}$$

During the iterative computation of f, the new term activates only when $f(x)$ has a negative value at x, or has a local minimum where v is very small. The two conditions, in fact, illustrate cases that are impermissible during the formulation of a PDM. The new term causes f to escape

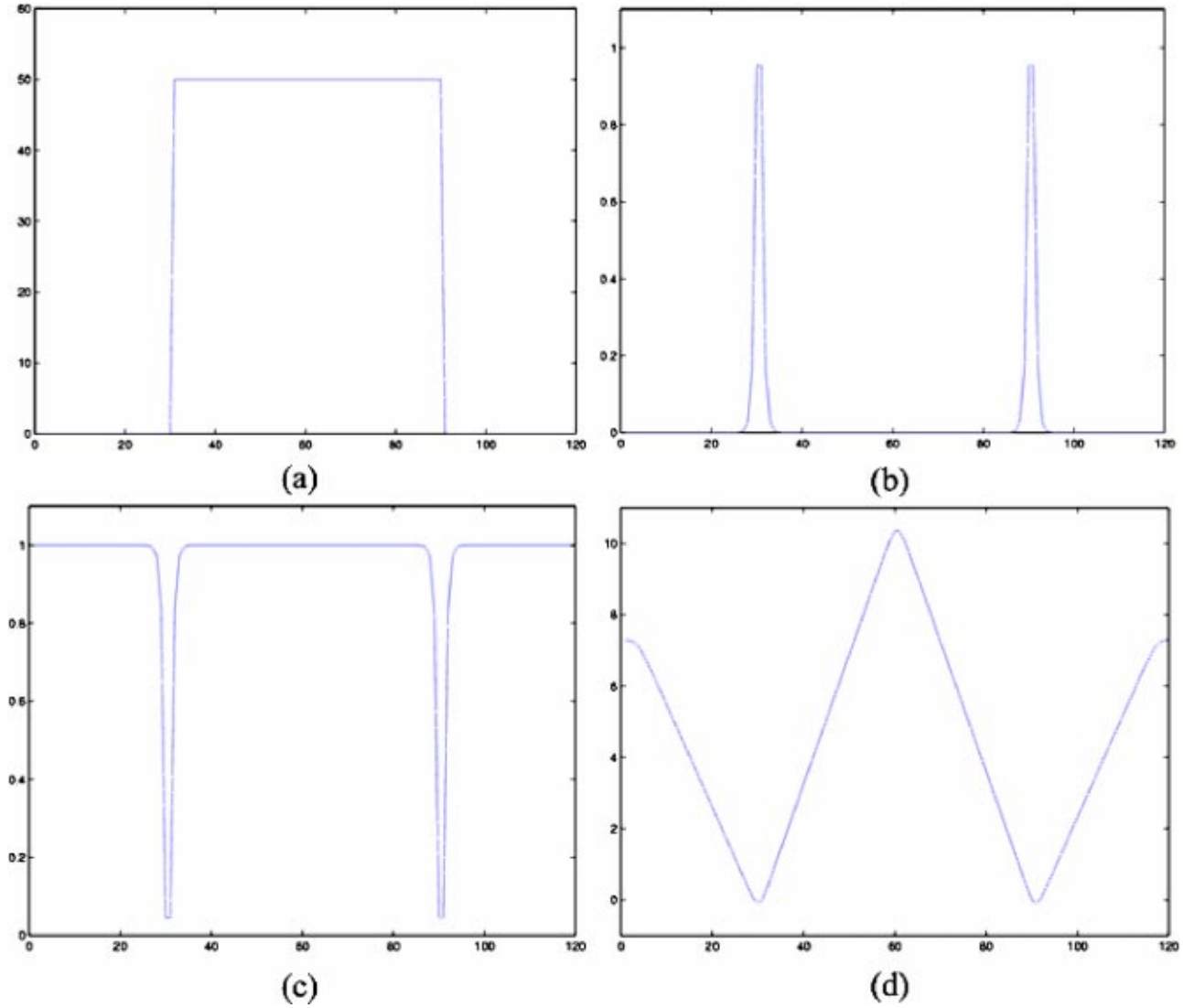


Fig. 2 (a) Input signal, (b) ESF v, (c) initial $f=1-v$, and (d) f after 60,000 iterations.

from them by smoothing it out where they occur. Our actual implementation of ϕ is as follows:

$$\phi=\begin{cases}1 & \text{if } \{f<\epsilon_1\} \quad \text{or } \{|g|<\epsilon_2, g_x>\epsilon_3, v<\epsilon_4\},\\ 0 & \text{otherwise,}\end{cases} \tag{17}$$

where $\epsilon_1 \leqslant 0, \epsilon_2>0, \epsilon_3>0, \epsilon_4>0$, and their absolute values are very small.

An illustrative example is given in Fig. 2. The input signal of Fig. 2(a) has two step edges at $x=30$ and 90. The ESF of Fig. 2(b) was obtained from the input signal using Ambrosio and Tortorelli's method, whose details are given in Sec. 2.2. The result of Fig. 2(d) was obtained by setting $\alpha=0.1$, $\beta=1.0$, $a=0.35$, $b=3a$, $\gamma=3.0$, $\epsilon_1=-0.1$, $\epsilon_2=0.05$, $\epsilon_3=0.01$, $\epsilon_4=0.05$, and $\Delta t=0.1$. Note that we can compute the approximate distance between the two edges from the value of f at the center. In Fig. 2(d), f is equal to 10.35 at the center, from which the distance is computed by $10.35/a\times 2=59.14$, which is approximately equal to the true distance 60.

At first glance, our method seems to involve many parameters to be adjusted. However, considering the role of each parameter, one can see that the parameters γ, ϵ_1, ϵ_2, ϵ_3, ϵ_4, a, and b are independent of the given ESFs. Therefore, once the parameter values selected are proven to be suitable for convergence, they are, ipso facto, also suitable for other ESFs.

Note that just as in the case of the ESF, we can control the smoothness of the resulting PDM by adjusting α and β. As shown in Fig. 3, this characteristic is verified by detecting ridges (they actually correspond to skeletons of the given image) from PDMs with a different smoothness. It is obvious from the results that, as α/β becomes larger, more detailed structures appear, but the resulting ridges become noisier.

2.3.2 *Extension to two dimensions*

The extension of Eq. (12) to two dimensions is straightforward:

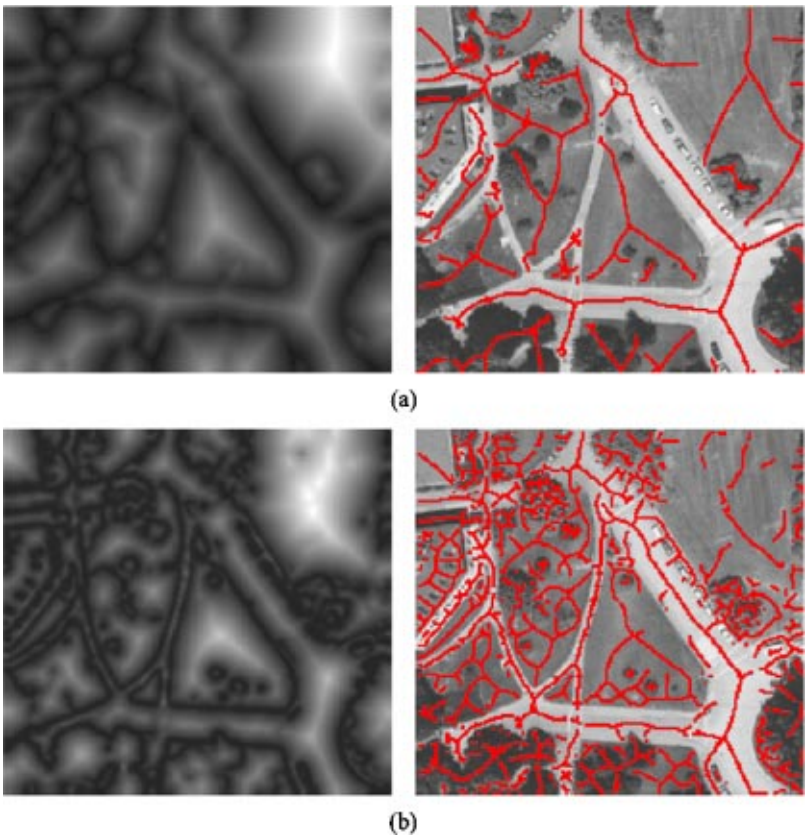


Fig. 3 Ridges of PDMs obtained from the input image of Fig. 10 (shown in Sec. 4) with different values of α and fixed value of $\beta=1.5$: (a) $\alpha=0.02$ and (b) $\alpha=0.2$.

$$E(f,g,h)=\int \alpha v\Big[f^2+f_x^2+f_y^2+(b-g_x)^2+\frac{1}{2}(g_y+h_x)^2 + (b-h_y)^2\Big]+\beta(a^2-\|\nabla f\|^2)^2+(g-f_x)^2 + (h-f_y)^2 \; dx\, dy, \tag{18}$$

where

$$g\approx f_x \quad \text{and} \quad h\approx f_y. \tag{19}$$

Consequently,

$$f_{xx}\approx g_x, f_{yy}\approx h_y, \text{ and } f_{xy}\approx\frac{1}{2}(g_y+h_x). \tag{20}$$

The corresponding PDEs are

$$f_t=\alpha(\nabla\cdot v\nabla f-vf)+2\beta[(\|\nabla f\|^2-a^2)\nabla^2 f+2(f_x^2 f_{xx} + 2f_x f_y f_{xy}+f_y^2 f_{yy})]+(\nabla^2 f-g_x-h_y), \tag{21}$$

$$g_t=\alpha\{2[v(g_x-b)]_x+[v(g_y+h_x)]_y\}+2(f_x-g), \tag{22}$$

$$h_t=\alpha\{[v(g_y+h_x)]_x+2[v(h_y-b)]_y\}+2(f_y-h). \tag{23}$$

We also added a new term $\gamma\phi\nabla^2 f$ to Eq. (21) for faster and more reliable convergence, where ϕ is given by

$$\phi=\begin{cases}1 & \text{if } \{f<\epsilon_1\} \text{ or } \{|\det(\mathbf{H})|<\epsilon_2, \text{trace}(\mathbf{H})>\epsilon_3, v<\epsilon_4\}\\ 0 & \text{otherwise,}\end{cases} \tag{24}$$

where $\epsilon_1\leq 0$, $\epsilon_2>0$, $\epsilon_3>0$, $\epsilon_4>0$, and their absolute values are very small. The matrix $\mathbf{H}$ is a Hessian matrix of f, which is given by

$$\mathbf{H}=\begin{bmatrix}f_{xx} & f_{xy}\\ f_{xy} & f_{yy}\end{bmatrix}\approx\begin{bmatrix}g_x & 1/2\,(g_y+h_x)\\ 1/2\,(g_y+h_x) & h_y\end{bmatrix}. \tag{25}$$

The condition $\{|\det(\mathbf{H})|<\epsilon_2,\ \text{trace}(\mathbf{H})>\epsilon_3\}$ in Eq. (24) is provided to detect valleys (i.e., local minimum) of f. Note that the two eigenvalues λ_1 and λ_2 ($|\lambda_1|\leq|\lambda_2|$) of the Hessian matrix of $f(x,y)$ correspond, respectively, to the minimum and maximum second-order directional derivatives (i.e., curvatures) of f at (x,y). Therefore, at the valleys of f, the following condition is usually satisfied:

$$\det(\mathbf{H})=\lambda_1\cdot\lambda_2\approx 0 \quad \text{and} \quad \text{trace}(\mathbf{H})=\lambda_1+\lambda_2\approx\lambda_2>0. \tag{26}$$

2.3.3 *Numerical implementation*

The PDEs introduced in this paper are numerically solved using the following finite-difference approximations:[17]

$$\frac{\partial f}{\partial t} \approx \frac{1}{\Delta t}[f(t+\Delta t)-f(t)], \tag{27}$$

$$\frac{\partial f}{\partial x} \approx \frac{1}{2\Delta x}[f(x+\Delta x)-f(x-\Delta x)], \tag{28}$$

$$\frac{\partial^2 f}{\partial x^2} \approx \frac{1}{\Delta x^2}[f(x+\Delta x)-2\,f(x)+f(x-\Delta x)], \tag{29}$$

$$\frac{\partial^2 f}{\partial x \partial y} \approx \frac{1}{4\Delta x \Delta y}[f(x+\Delta x,y+\Delta y)-f(x-\Delta x,y+\Delta y)] - f(x+\Delta x,y-\Delta y)+f(x-\Delta x,y-\Delta y)]. \tag{30}$$

Actually, the space steps Δx and Δy were fixed to 1 throughout all the experiments. The time step Δt must be adjusted for the convergence of the PDEs.

2.3.4 *Reduction of computational cost*

As mentioned, we obtain a PDM by numerically solving the corresponding PDEs using finite-difference methods (FDMs). However, these schemes are stable only for very small time steps, which reduces efficiency and results in a large computational cost. For example, in the case of Fig. 10 (182×174) in Sec. 4, it takes about 25 min to obtain a PDM after 60,000 iterations by directly solving PDEs using an FDM. This computational time may limit the practical usage of the proposed method. By a simple and heuristic

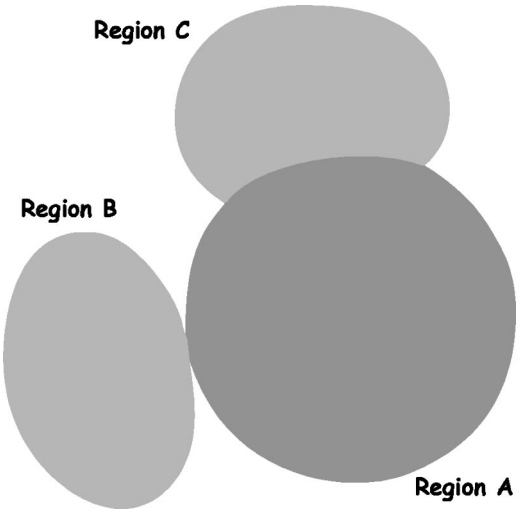


Fig. 4 Illustration of determining a weighting factor for the combination of two merging criteria.

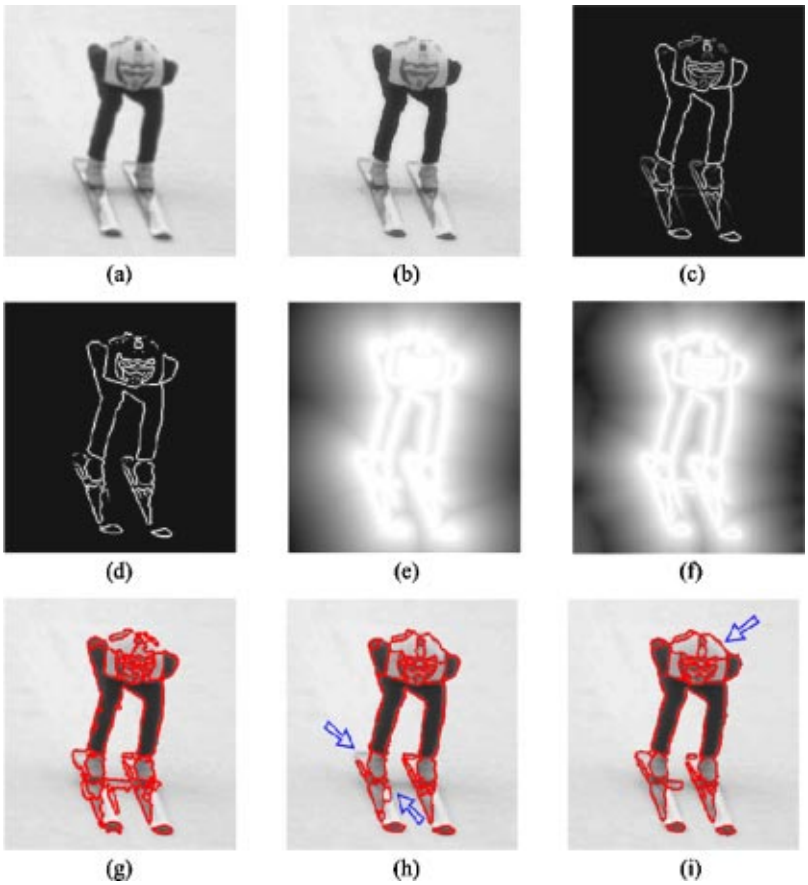


Fig. 5 Segmentation results of the gradient-, EDM-, and PDM-based methods: (a) original image, (b) noise-reduced image, (c) ESF obtained from (b), (d) edge image by thresholding (c), (e) inverted EDM obtained from (d), (f) inverted PDM obtained from (c), (g) final segmentation result based on thresholded gradients proposed by Haris *et al.*, (h) final segmentation result based on the EDM, and (i) final segmentation result based on the PDM.

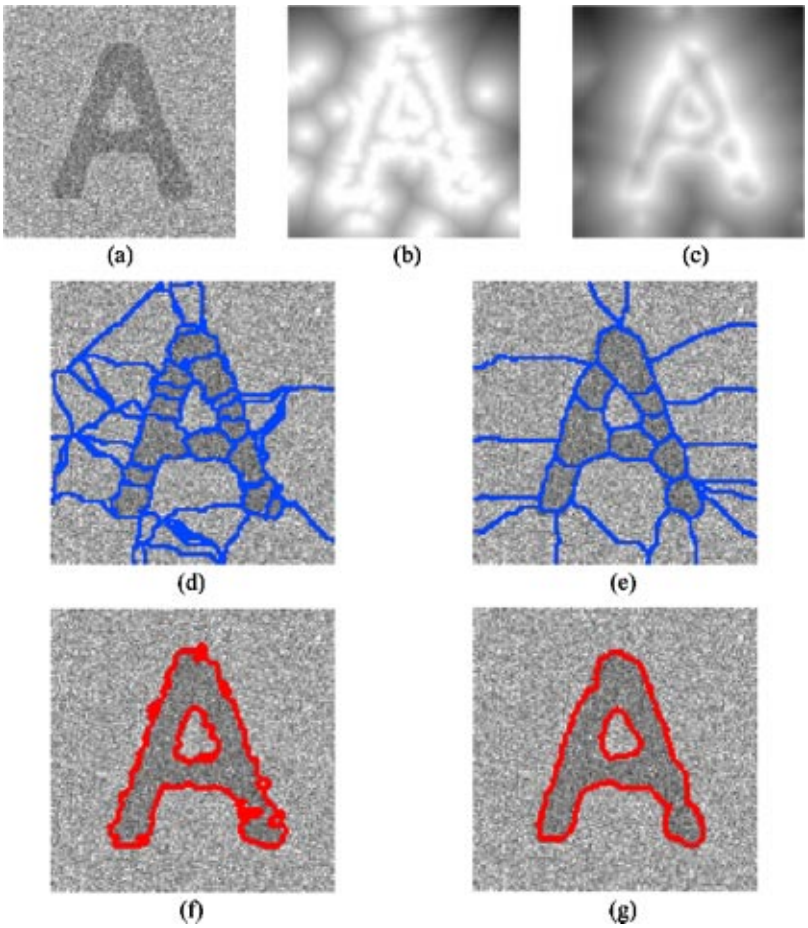


Fig. 6 Comparison between the EDM- and PDM-based methods: (a) original image, (b) inverted EDM, (c) inverted PDM, (d) watersheds of (b), (e) watersheds of (c), (f) region-merging result of (d), and (g) region-merging result of (e).

idea, we reduce the computational time as follows. At the current iteration, not all the pixels have meaningful changes compared with previous iteration. Therefore, it is not necessary to evaluate the PDEs of Eqs. (21), (22), and (23) involving the first or second derivatives at all pixel positions or at every iteration. Hence, at each iteration, we evaluate these PDEs only at the pixels with noticeable changes over previous iterations and their 8 neighborhood pixels. When the number of pixels with noticeable changes is less than a given threshold, the computation of the PDM is stopped. In this manner, in Fig. 10 in Sec. 4 we can reduce the computational time of 25 min to about 6 min. In addition, if we give a more sophisticated initial value of a PDM instead of the opposite of an ESF, we can reduce the computational time much further. We propose a hierarchical approach, which first calculates a down-sampled PDM with an initial value of the opposite of a down-sampled ESF and next obtains a final PDM with an initial value of an up-sampled version of the calculated PDM. Using this approach, we can obtain a PDM of the same quality as a standard FDM after only 20,000 iterations, so the computational time is reduced to 3.5 min.

3 Robust Image Segmentation Based on a PDM

In this section, we introduce the overall procedure of our region segmentation method along with our new region-merging criterion. The procedure consists of the following four steps:

1. An ESF is computed from an input gray-scale image using the method described in Sec. 2.2.
2. A PDM is computed from the ESF by numerically solving the PDEs introduced in Sec. 2.3.
3. An initial region segmentation result is obtained by applying the watershed transformation to the inverted PDM. We used Vincent and Soille's watershed algorithm.[5]
4. Region merging is carried out with our new merging criterion.

There are several region-merging criteria proposed in the literature. For example, Haris *et al.*[18] proposed a dissimilarity function based on average intensities of two adjacent regions and their associated areas. In Ref. 19, standard deviations of region intensities are considered, as well

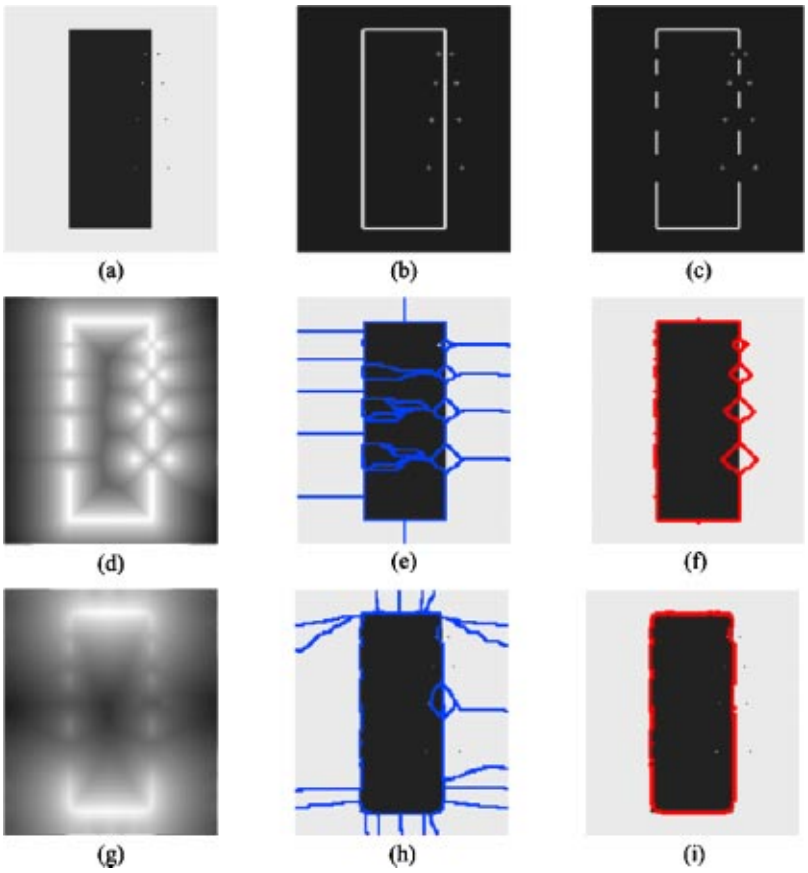


Fig. 7 Other segmentation results of the EDM- and PDM-based methods: (a) original image, (b) ESF obtained from (a), (c) modified ESF, (d) inverted EDM obtained from (c), (e) watersheds of (d), (f) region-merging result of (e), (g) inverted PDM obtained from (c), (h) watersheds of (g), and (i) region-merging result of (h).

as their means. However, according to our experiments, these criteria often produced unsatisfactory merging results. It was observed that the main reason for the failure was that no edge strength information along the common boundary was considered in those criteria. Moreover, in the case of two large adjacent regions, the criterion of Haris *et al.* was dominantly affected by their areas regardless of their average intensities, which consequently caused poor merging results.

Motivated by this observation, we developed a new dissimilarity function appropriately combining edge strength and region intensity information. Let R_i represent the set of pixels belonging to a region i, and let Γ_i and Γ_{ij} represent the set of boundary pixels of the region i and the set of pixels belonging to the common boundary between the regions i and j, respectively. If the two regions i and j are adjacent to each other and $\|R_i\| \leqslant \|R_j\|$, where $\|R\|$ represents the cardinality of a set R, the proposed function measuring the dissimilarity between the regions i and j is given by

$$\delta(i,j) = \kappa \frac{\|\Gamma_{ij}\|}{\|\Gamma_i\|} E(\Gamma_{ij}) + \left(1 - \frac{\|\Gamma_{ij}\|}{\|\Gamma_i\|}\right) |\mu(R_i) - \mu(R_j)|, \tag{31}$$

where $E(R)$ and $\mu(R)$ are the average edge strength and intensity values of pixels belonging to R, and κ is a scaling constant. In this equation, the ratio $r = \|\Gamma_{ij}\|/\|\Gamma_i\|$ is used as a weighting factor to combine $E(\Gamma_{ij})$ and $|\mu(R_i) - \mu(R_j)|$. The reason for this can be well understood by considering Fig. 4, where the regions B and C have the same intensities and the common boundary of A and C is much longer than that of A and B. In the case of the regions A and C (large r), the edge strength information on the common boundary seems more reliable than the region intensity information. Therefore, when trying to merge regions A and C, we should emphasize $E(\Gamma_{AC})$ more than $|\mu(R_A) - \mu(R_C)|$, which is the opposite of the case of A and B (small r). Thus, the ratio r is suitable for determining the weights of $E(\Gamma_{ij})$ and $|\mu(R_i) - \mu(R_j)|$.

4 Experimental Results

To show the usefulness of our method, we compared the performance of our method with the gradient-based method proposed by Haris *et al.*[18] and the EDM-based method. The method of Haris *et al.* consists of four stages. The first stage is to reduce the noise that corrupts an original image while preserving its meaningful structures. At the second stage, Gaussian gradients are calculated and their magni-

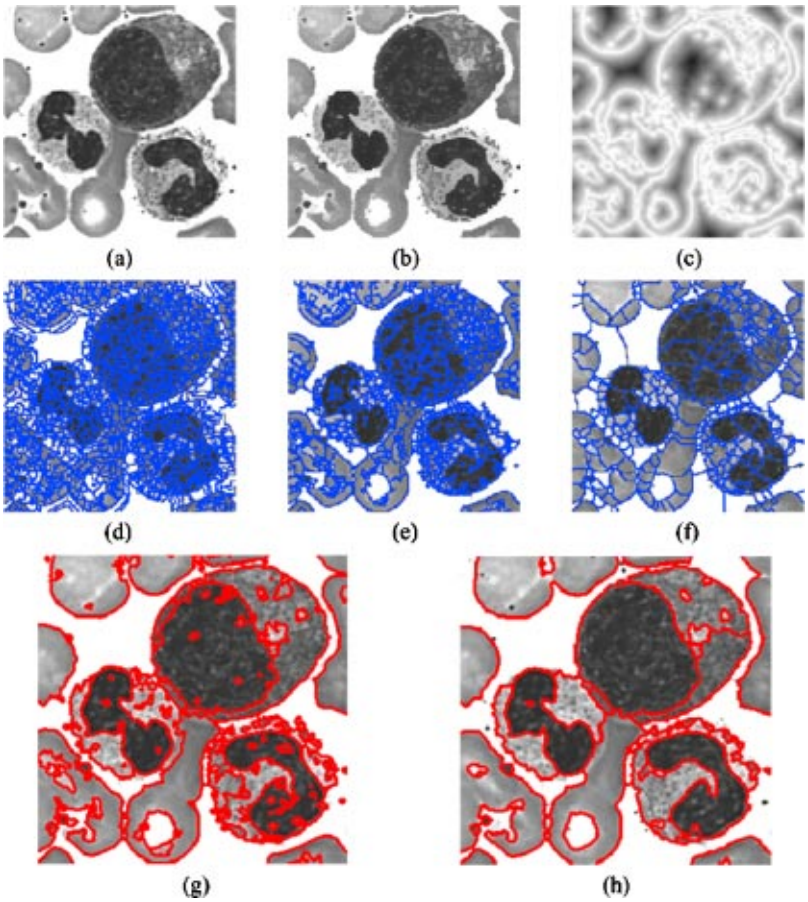


Fig. 8 Comparison between the gradient- and PDM-based methods: (a) original image, (b) noise-reduced image, (c) inverted PDM obtained from the ESF of (b), (d) watersheds of gradients (2531 regions), (e) watersheds of thresholded gradients (1639 regions), (f) watersheds of the inverted PDM (372 regions), (g) region-merging result of (e), and (h) region-merging result of (f).

tudes below a certain threshold are set to zero. At the next stage, the resulting thresholded gradient image is fed into the watershed algorithm, which produces an initial segmentation result. At the final stage, an iterative region-merging algorithm is applied to the watershed regions to produce a final segmentation result. For a fair comparison, we applied the same noise reduction technique proposed by Haris *et al.* to an input image before obtaining the ESF or Gaussian gradients. We also used the same watershed algorithm proposed by Vincent and Soille[5] and the same dissimilarity function of Eq. (31), where we normalized the gradient magnitudes to use the dissimilarity function in the method of Haris *et al.* Further, using the region-growing Euclidean distance transform algorithm proposed by Cuisenaire,[20] we compute the EDM from the edge image, which is obtained by thresholding the ESF. Therefore, the largest difference among the methods used in these experiments is the input to the watershed transformation. In our method, the PDM is used instead of the thresholded gradient image or the EDM.

Figure 5 shows the segmentation results of the gradient-, EDM-, and PDM-based methods. The ESF of Fig. 5(c) was obtained by Ambrosio and Tortorelli's method of Sec. 2.2, with $\mu=3.5$ and $\eta=3.0$, and the PDM of Fig. 5(f) was obtained from the ESF, with $\alpha=0.6$ and $\beta=1.0$. The final segmentation results of Figs. 5(g), 5(h), and 5(i) were obtained by iteratively merging adjacent regions whose dissimilarity is less than 15.0. Comparing the three figures, one can easily recognize that the proposed method produced a more accurate and meaningful result with less noisy segmentation. Note that the part indicated by an arrow in Fig. 5(i) clearly shows that our method can provide well-completed region boundaries even if some parts of the boundary have nearly zero edge strength values. This ability to detect meaningful shape boundaries in a gray-scale image can be regarded as an analogy of the binary image case shown in Fig. 1. The result of the EDM-based method in Fig. 5(h) also produces well-completed region boundaries as does our method. However, since the EDM is computed from the edge image of Fig. 5(d), which was obtained by thresholding the ESF of Fig. 5(c), its segmentation result is more likely to be noisy due to spurious edges as well as more likely to lose useful boundary information of input regions [see the arrowed parts in Fig. 5(h)].

Figure 6 shows the robustness of the PDM to the noise in comparison with the EDM. Figure 6(a) is a synthetic image corrupted by additive Gaussian noise. The inverted EDM of Fig. 6(b) was obtained by applying the Euclidean distance transform to the edge image obtained by thresholding the ESF computed from the input image of Fig. 6(a). We can observe from Figs. 6(f) and 6(g) that the PDM-

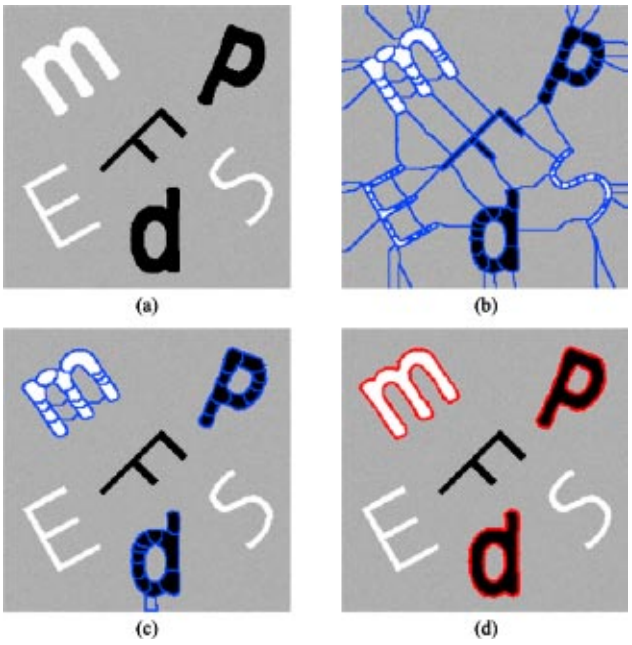


Fig. 9 Selective region merging using the PDM information on the width of regions: (a) original image, (b) watersheds of the inverted PDM, (c) selected watershed regions of specific widths (10 to 25 pixels), and (d) region-merging result of (c).

based method is less sensitive to noise than the EDM-based method.

Figure 7 shows the robustness of the PDM-based method more clearly. Figure 7(a) shows an image of a rectangular object with black and white dots along its boundary. The ESF of Fig. 7(b) was computed from the image of Fig. 7(a). We obtained the image of Fig. 7(c) from the ESF of Fig. 7(b) by intentionally making holes (5 to 20 pixels wide) along the object boundary. From Figs. 7(f) and 7(i), we can see that our method correctly extracted the rectangular object in spite of the existence of large holes and strong noises around the holes, while the EDM-based method failed to find the correct region boundary due to its sensitivity to noise.

From the results of Fig. 8, we can see that, although the method of Haris *et al.* significantly reduces the number of initial partitions by applying the watershed transformation to a thresholded gradient image, it still produces a large number of initial partitions when compared to our method. In addition, the final segmentation results show that the proposed PDM-based segmentation method provides region boundaries that are less noisy and more accurate than the method using thresholded gradients.

Moreover, by utilizing information on the width of regions contained in the proposed PDM, our segmentation method can selectively extract regions of specified width, as illustrated in Fig. 9. Figure 9(c) was obtained by selecting watershed regions whose maximum PDM values range from 10 to 25 pixels. Figure 9(d) was obtained by iteratively merging the selected regions.

Other examples of the PDM-based segmentation are shown in Fig. 10. The gradients computed in these general images can also be noisy and therefore the gradient-based or the EDM-based methods using gradient information can present poor results. However, as you see in the experimental results, the proposed method shows good segmentation performance, which is promising and robust to noise.

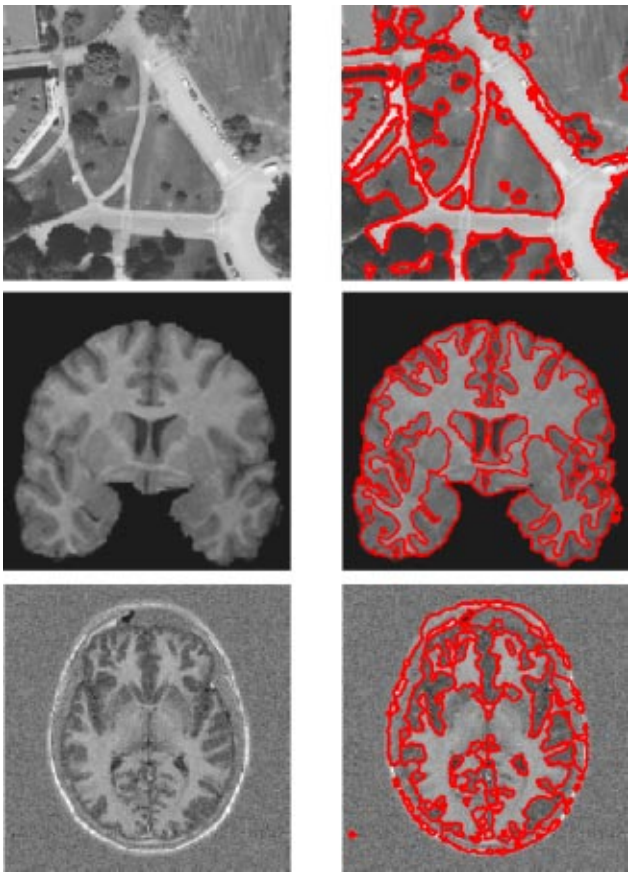


Fig. 10 Other PDM-based segmentation results.

5 Conclusions

We proposed a new region segmentation method based on the watershed transformation of a PDM. Since the PDM is a regularized version of a EDM and is directly computed from an ESF, its watershed transformation usually produces more reliable and less noisy initial segmentation results than gradient-based or EDM-based methods, while preserving the useful property of the EDM-based method. We also proposed an efficient region-merging criterion based on edge strengths and region intensities. Through several experiments we showed that the proposed region-merging criterion, along with the PDM-based method, yielded final segmentation results superior to conventional methods.

The main drawback of our method is that it requires a long computation time. As represented in Sec. 2.3.4, we attempted to speed up our method with a heuristic idea and an accurate initialization. Nevertheless, the proposed method still has a computational burden. Weickert *et al.*[21] proposed efficient and reliable schemes for nonlinear diffusion filtering using additive operator splitting (AOS). By adopting this scheme for a future work, we will try to further reduce the computational cost of our method at least 10 times (a further speed-up by a hundred times is possible by implementing on a parallel computer), which will lead our method to various practical usages.

As another future work, we would like to adjust the PDM parameters according to the characteristics of the given image. Especially, as shown in Fig. 3, it will be very nice if we can automatically control the ratio of two constraint terms—(A) and (B) in Eq. (11)—by adjusting α and β according to the details of image contents or applications. Further, we can also improve our work by automatically determining the ESF parameters μ and η.

Finally, as a segmentation tool we can adopt one of more sophisticated methods such as geodesic active contours. Since the proposed method is based on the watershed algorithm, it seems to be a simple method but experimental results are very promising and well represent its robust properties. After careful consideration of initializations and parameter tunings, if we use a more advanced method, it will be a new segmentation approach compared with the proposed method.

Acknowledgments

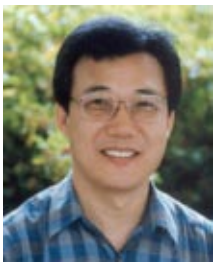
The authors would like to thank for the financial support of the Ministry of Education of Korea toward the Electrical and Computer Engineering Division at POSTECH through its Brain Korea 21 (BK21) program. They also thank the anonymous reviewers for their helpful comments, which improved the quality of this paper.

References

1. H. Blum, ''A transformation for extracting new descriptions of shape,'' in *Models for Perception of Speech and Visual Form*, pp. 362–380, MIT Press, Cambridge, MA (1967).
2. C. Arcelli and G. S. di Baja, ''Ridge points in Euclidean distance maps,'' *Pattern Recogn. Lett.* **13**, 237–243 (1992).
3. G. Malandain and S. Fernández-Vidal, ''Euclidean skeletons,'' *Image Vis. Comput.* **16**, 317–327 (1998).
4. L. Vincent and S. Beucher, ''The morphological approach to segmentation: an introduction,'' Internal Rep. CMM, School of Mines, Paris (1989).
5. L. Vincent and P. Soille, ''Watersheds in digital spaces: an efficient algorithm based on immersion simulations,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(6), 583–598 (1991).
6. J. H. Jang and K. S. Hong, ''A pseudo-distance map for the segmentation-free skeletonization of gray-scale images,'' in *Proc. Int. Conf. on Computer Vision*, Vol. 2, pp. 18–23, IEEE, Piscataway, NJ (2001).
7. J. A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, Cambridge (1999).
8. K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker, ''The Hamilton-Jacobi skeleton,'' in *Proc. Int. Conf. on Computer Vision*, Vol. 2, pp. 828–834, IEEE, Piscataway, NJ (1999).
9. P. Perona and J. Malik, ''Scale-space and edge detection using anisotropic diffusion,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(7), 629–639 (1990).
10. A. Bovik, Ed., *Handbook of Image and Video Processing*, pp. 433–447, Academic Press, San Diego, CA (2000).
11. B. Jähne, H. Haußecker, and P. Geißler, *Handbook of Computer Vision and Applications*, Vol. 2, pp. 423–450, Academic Press, San Diego, CA (1999).
12. L. Ambrosio and V. M. Tortorelli, ''Approximation of functionals depending on jumps by elliptic functionals via γ-convergence,'' *Boll. Unione Mat. Ital.* **7**, 105–123 (1992).
13. B. K. P. Horn, ''The calculus of variations,'' in *Robot Vision*, pp. 469–474, MIT Press, Cambridge, MA (1986).
14. J. Shah, ''A common framework for curve evolution, segmentation and anisotropic diffusion,'' in *Proc. Computer Vision and Pattern Recognition*, Vol. 1, pp. 136–142, IEEE, Piscataway, NJ (1996).
15. S. Teboul, L. Blanc-Féraud, G. Aubert, and M. Barlaud, ''Variational approach for edge-preserving regularization using coupled PDE's,'' *IEEE Trans. Image Process.* **7**(3), 387–397 (1998).
16. M. Proesmans, E. Pauwels, and L. van Gool, ''Coupled geometry-driven diffusion equations for low-level vision,'' in *Geometry-Driven Diffusion in Computer Vision*, B. M. ter Haar Romeny, Ed., pp. 191–228, Kluwer Academic (1994).
17. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, ''Numerical derivatives,'' in *Numerical Recipes in C*, pp. 186–189, Cambridge University Press, Cambridge (1992).
18. K. Haris, S. N. Efstratiadis, N. Maglaveras, and A. K. Katsaggelos, ''Hybrid image segmentation using watersheds and fast region merging,'' *IEEE Trans. Image Process.* **7**(12), 1684–1699 (1998).
19. I. Pitas, ''Image segmentation algorithms,'' in *Digital Image Processing Algorithms and Applications*, pp. 254–297, Wiley Publishing, Inc., Indianapolis, IN (1993).
20. O. Cuisenaire, ''Region growing Euclidean distance transforms,'' in *Proc. Int. Conf. on Image Analysis and Processing*, Vol. 1, pp. 263–270, IAPR (1997).
21. J. Weickert, B. M. ter H. Romeny, and M. A. Viergever, ''Efficient and reliable schemes for nonlinear diffusion filtering,'' *IEEE Trans. Image Process.* **7**(3), 398–410 (1998).

Byoung-Ki Jeon received his BS degree in electronic engineering in 1997 from Kyungpook National University, Korea, and his MS degree in electrical and electronic engineering in 1999 from POSTECH, Korea, where he is now in the PhD program. His research interests include synthetic aperture radar image processing, image segmentation, and machine learning.

Jeong-Hun Jang received his BS degree in electrical engineering from Hanyang University, Korea, in 1994 and his MS and PhD degrees in electrical and electronic engineering from POSTECH, Korea, in 1996 and 2001, respectively. He is currently a CTO with Illisis Co. Ltd., Korea. His current research interests include panoramic image and video creation, image-based 3-D modeling, and augmented reality.



Ki-Sang Hong received his BS degree in electronic engineering from Seoul National University, Korea, in 1977, his MS degree in electrical and electronic engineering in 1979, and his PhD degree in 1984 from KAIST, Korea. From 1984 to 1986 he was a researcher with the Korea Atomic Energy Research Institute and in 1986 he joined POSTECH, Korea, where he is currently a professor with the Division of Electrical and Computer Engineering. From 1988 to 1989 he was a visiting professor with the Robotics Institute at Carnegie Mellon University, Pittsburgh, Pennsylvania. His current research interests include computer vision, augmented reality, pattern recognition, and synthetic aperture radar image processing.