Image and Vision Computing xxx (2008) xxx–xxx

Contents lists available at ScienceDirect

Image and Vision Computing

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

An extension of min/max flow framework

Hongchuan Yu [a,*], Mohammed Bennamoun [b], Chua Chin-Seng [c]

[a] The Media School, Bournemouth University, Poole, UK
[b] School of Computer Science and Software Engineering, University of Western Australia, Perth, WA 6009, Australia
[c] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore

ARTICLE INFO

Article history:
Received 30 January 2004
Received in revised form 15 September 2006
Accepted 7 May 2008
Available online xxxx

Keywords:
Q1 Min/max flow framework
Anisotropic diffusion
Boundary leaking
Image segmentation
Region tracking

ABSTRACT

In this paper, the min/max flow scheme for image restoration is revised. The novelty consists of the following three parts. The first is to analyze the reason of the speckle generation and then to modify the original scheme. The second is to point out that the continued application of this scheme cannot result in an adaptive stopping of the curvature flow. This is followed by modifications of the original scheme through the introduction of the Gradient Vector Flow (GVF) field and the zero-crossing detector, so as to control the smoothing effect. Our experimental results with image restoration show that the proposed schemes can reach a steady state solution while preserving the essential structures of objects. The third is to extend the min/max flow scheme to deal with the boundary leaking problem, which is indeed an intrinsic shortcoming of the familiar geodesic active contour model. The min/max flow framework provides us with an effective way to approximate the optimal solution. From an implementation point of view, this extended scheme makes the speed function simpler and more flexible. The experimental results of segmentation and region tracking show that the boundary leaking problem can be effectively suppressed.

1. Introduction

Linear and non-linear Partial Differential Equations (PDEs) models have been applied to image restoration and analysis for about two decades now. Of interest to PDEs models are studies related to image selective smoothing, whereby the smoothing is adaptively controlled not only by the amount of smoothing but also by the direction along the image features. The classic models include the Perona–Malik equation [1] and the mean curvature flow [7], which smooth out noise or trivial textures in an image while preserving the essential structures or boundaries of the object. There exists an extensive literature [1–10] which addresses both their theoretical and application aspects, wherein the mean curvature flow is one of the most popular anisotropic diffusion models. Alvarez et al. in [3] proved that the mean curvature was invariant under changes of illumination, positions, orientations and scales of objects. Lu et al. in [15] further studied the evolutional behavior of the mean curvature flow and the two principal curvature flows. Indeed, the attractive quality of the mean curvature flow model is that sharp boundaries are preserved, i.e. smoothing takes place inside a region, but not across region boundaries. But then, due to Grayson's theorem [4], it is known that each contour shrinks to zero and disappears through the continued application of the curvature flow scheme. Consequently, a stopping criterion is required. In order to control the model's evolution, Malladi and Sethian in [5,6] devised a min/max flow scheme for image enhancement and denoising under a level set numerical framework. Their basic idea is to correctly select the evolving curvature flow using the min/max switch function to remove noise or to enhance the image. Unfortunately, the implementation of this scheme is accompanied with a numerical drawback which results in the apparition of some speckle in the grey-level (or color) image and the contamination of the entire image gradually. An example is shown in Fig. 2(b–d). In addition, the continued application of this scheme cannot also result in the adaptive stopping of the curvature flow, i.e. the evolving flow cannot stop even when noise is removed. The larger and global properties of the shapes in an image are also smoothed out with this scheme. However, this approach remains valuable and this paper will present ways on how to overcome its drawbacks. Furthermore, based on the min/max scheme, the zero-crossing detector of the second order derivative and the Gradient Vector Flow (GVF) field [11] are introduced in our modified schemes, so that the min/max flow can adaptively stop once the noise is removed.

The min/max flow scheme is a flexible computational framework, and many methods and strategies can indeed be integrated into this framework. At present, the geodesic active contour model is widely applied to image segmentation and region tracking applications. However boundary leaking is still a challenging problem, which is indeed related to the stopping criterion. Some constraint

* Corresponding author. Tel.: +61 8 64883222.
E-mail addresses: cnyuhc@yahoo.com (H. Yu), m.bennamoun@csse.uwa.edu.au (M. Bennamoun), ECSChua@ntu.edu.sg (C. Chin-Seng).

0262-8856/$ - see front matter Crown Copyright © 2008 Published by Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2008.05.006

terms have been introduced into the geometric active contour model. This includes the weighted area gradient flow [12], the gradient vector flow [11,13], and the edge-flow [14]. However, it is possible, as demonstrated below, to suppress the boundary leaking if the evolving flow could be controlled. Indeed the min/max switch function in the min/max flow scheme provides us with an effective manner to correctly select an evolution equation. In this paper, we will further extend the min/max scheme and apply it to deal with the boundary leaking problem. A straightforward advantage of this extension scheme is that the expressions of the speed function for image segmentation and region tracking are similar, while their different decision rules (boundary conditions) can be defined, respectively. This makes the design of the speed function simpler and more flexible for various different applications.

Different from the geodesic active model, Chan and Vese in [16,17] recently presented other kinds of active contour models through the introduction of the partition thresholding under a minimal variance criterion. These models do not usually need a stopping term which is based on the image gradient in their evolution equation. However, in this paper, we still pay attention to the familiar geodesic active contour model. This is because it is a prototypical model in many applications. Indeed, the partition thresholding approach can also be seen as a constraint term and can be introduced into the geodesic active model as described in [18].

This paper is organized as follows: the min/max flow scheme is first briefly introduced in Section 2. Then, the image anisotropic diffusion is analyzed under this scheme in detail in Section 3. The numerical drawback from the original scheme is demonstrated, and our presented schemes are also provided in this section. Section 4 extends the min/max flow framework to deal with the boundary leaking in image segmentation and region tracking applications. Finally, our conclusions appear in Section 5.

2. Min/max flow scheme

The min/max flow scheme was first introduced in [6] for the grey-scale, texture and color image enhancement and noise removal. An image is interpreted as a collection of iso-intensity contours which can be evolved. The level set equation of an image intensity I can be written as,

$$I_t = F|\nabla I|, \tag{1}$$

where the speed function is defined as

$$F(\mathbf{A}) = \begin{cases} \max(\kappa, 0), & \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) < \text{Threshold} \\ \min(\kappa, 0), & \text{otherwise} \end{cases},$$

$\mathbf{A},\mathbf{X} \in R^2$, $\Omega(\mathbf{A})$ is a neighborhood around some point A, κ is the curvature of the iso-intensity contour, and t is the evolving time. The above definition of the speed function is called the min/max flow framework. Consider a non-convex region $\mathbf{R}$ bounded by a closed iso-intensity contour, and denote by "inside" the region on the darker side (lower brightness values). For binary images, the threshold can be simply taken as the average of the two intensity values; while, for grey-scale or texture images, it can be estimated as the average value of the intensity obtained along the direction perpendicular to the gradient direction in the neighborhood Ω. The max flow shrinks the outward convexities of $\mathbf{R}$ until it becomes a small convex region, which then collapses to a point. The min flow, on the other hand, inflates the inward convexities of $\mathbf{R}$ until it becomes the convex hull of the starting region, thereafter diffusions stops. The particular behavior of the max flow and min flow are, respectively, summarized in the following properties (see [5,6] for details).

Property 1. The flow under $F = \min(\kappa,0)$ allows the inward concave fingers to grow outward, while suppressing the motion of the outward convex regions. Furthermore the motion halts as soon as a convex hull is obtained.

Property 2. The flow under $F = \max(\kappa,0)$ allows the outward convex regions to grow inward while suppressing the motion of the inward concave regions. Once the shape becomes fully convex, the flow becomes the same as a regular curvature flow, in which case the shape collapses to a point.

In fact, the key idea of the min/max flow scheme is to correctly select the evolving flow that both smoothes out small oscillations and maintains the essential details of the shape.

3. Image restoration under min/max flow framework

3.1. Analysis and solution of the speckle problem

The speckle problem appears in the implementation of the min/max flow scheme described in Eq. (1), and as a result of the presence of texture or noise in images. Through the following analysis, it can be shown that it is indeed an artifact caused by numerical instability.

First, let us imagine the binary case where there are only two grey levels in an image: one corresponds to the object intensity, and the other corresponds to the background intensity. The threshold in Eq. (1) is defined as the average of these two intensity values. Let us assume that the background color is lighter than the object color. Let some intensity contour pass through point A, and the circle region is the neighborhood around point A as illustrated in Fig. 1. If the threshold is less than the intensity average in the neighborhood around point A, the max flow is selected (see Fig. 1(a)). In this case, the shape of the object is convex in the neighborhood, and the curvature of point A is positive. The speed is then greater than zero. The intensity contour moves therefore inward the object region until the average becomes less, and the "min" switch takes over. If the threshold is greater than the average, the min flow is selected (see Fig. 1(b)). In this case, the shape of the object is concave in the neighborhood, and the curvature of point A is negative. The speed is less than zero and the intensity contour moves outward the object region. On that basis, one can conclude that the binary case does not present any problem.

When the observed image is a grey-scale or textured image, the analysis is more complicated than the binary case. The threshold in Eq. (1) is defined as the average value of the intensity obtained along the tangential line, i.e. the tangent to the intensity contour at point A as illustrated in Fig. 1. When the threshold is greater than the average of the neighborhood of point A, the max flow is selected. In this case, the shape of the object is convex in the neighborhood, and the curvature of point A is positive. Obviously, the intensity of point A should be updated by a larger intensity value. Because of the spurious edges, noise or numerical errors, it is possible that the intensity value of point A becomes greater than the threshold. Thus, the updated intensity value of point A becomes

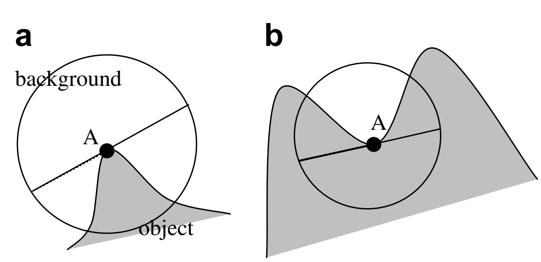


Fig. 1. The neighborhood and threshold of the min/max flow scheme.

greater than the intensity of all the neighboring points. After several iterations, the distinct highlight speckles will appear at point A. A similar analysis can be achieved in the case of dark speckles.

In Fig. 2(b–d), we illustrate the evolution results at different iterations. With a continued application of the scheme of Eq. (1), the speckles will contaminate the whole image gradually. In order to overcome this problem, the decision rule in Eq. (1) is revised. According to the two properties in Section 2, one can notice that the flow under $F = \max(\kappa, 0)$ diffuses away all of the information while the flow under $F = \min(\kappa, 0)$ preserves some of the structure. It is clear that when the average of the neighborhood is less than the threshold, the max flow should be selected so as to smooth out small oscillations, but not to enhance them. Thus, any enhancement operation (i.e. selecting the min flow) should be suppressed in this case. According to the above analysis, it is clear that the decision rule oversimplifies the choice between the max and min flows. On that basis, we re-define the min/max flow as,

$$F(\mathbf{A}) = \begin{cases} \max(\kappa, 0), & \text{if } \begin{cases} \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) < \text{Threshold} \\ I(\mathbf{A}) < \text{Threshold} \end{cases} \\ \min(\kappa, 0), & \text{if } \begin{cases} \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) > \text{Threshold} \\ I(\mathbf{A}) > \text{Threshold} \end{cases}, \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\Omega(\mathbf{A})$ is the neighborhood of point A. When the threshold is greater than the average, but the intensity of point A is greater than the threshold, the speed function is set to zero. In this case, no speckle appears at point A. An experimental illustration is shown in Fig. 2(e).In addition, we can also notice that some of the textures in the original image shown in Fig. 2(a) disappear as shown in Fig. 2(e) when the scheme of Eq. (2) is used.

3.2. Curvature equation under min/max flow framework

The curvature flow is one of the anisotropic diffusion models, which displays some particular geometric and numerical advantages. The most attractive quality is that smoothing takes place only inside a region, but not across region boundaries. According to Grayson's theorem [4], we know that all information is eventually removed through a continued application of the curvature flow scheme. In order to preserve some essential features after the continued application of the curvature flow scheme, the min/max scheme was introduced in the speed function in [5,6]. We will conclude that this expectation cannot be achieved under the scheme of Eq. (2) through the following analysis.

Roughly speaking, the curvature flow can lead Eq. (2) towards a harmonic solution. We are therefore assuming that the final solution (intensity function $I(X)$) is a harmonic function. According to the mean value theorem for harmonic functions, we know that the isolated noise points and notch shaped structure in the neighborhood around some point is smoothed out gradually. It can be verified that the inflectional shape or boundary in a neighborhood will gradually become straight or die out according to the definition of the speed function given in Eq. (2). To illustrate this point, we show the initial notched shapes in a neighborhood around some point in Fig. 3(a and b). The result using the scheme of Eq. (2) is shown in Fig. 3(c). It can be seen that the notched region can be smoothed out.

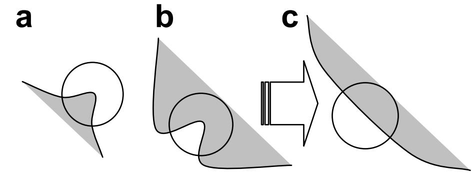


Fig. 3. Illustration that the boundary of the shapes in (a) and (b) are straightened out.

On that basis, we propose the following proposition,

Proposition 1. *The speed function in Eq. (2) results in the edges or boundaries of the shape to be straightened up.*

Now, let us consider the continued application of the scheme of Eq. (2). It is expected that all oscillations below some radius level be removed, while all features above that level are preserved, and the algorithm can stop automatically once the sub-scale noise is removed. In other words, after enough iteration steps, the inflection edge turns to a straight line in the neighborhood around every point that is on the shape boundary according to Proposition 1, while all the features above that radius are preserved. Unfortunately, this assumption cannot be achieved in practice. For convenience, this can further be explained as follows: Assume that L is a bending boundary above some radius r, and the circles denote the neighborhood around points x_1 and $x_n \in L$ (see Fig. 4). Let point x_1 be far away from point x_n. We assume that in the neighborhoods $|\Delta x| \leqslant r$ around he two points x_1 and x_n, the two segments of L are both straight lines, $\overline{A_1B_1}, \overline{A_nB_n} \subset L$, whose extension lines intersect at point C.

Further assume that L is piecewise continuous. For the adjacent point $\mathbf{x}_2$ of $\mathbf{x}_1$, the segment $\overline{A_2B_2}$ of L intercepted by the neighborhood $|\Delta \mathbf{x}| \leqslant r$ around $\mathbf{x}_2$ must be a straight line, and the slope of $\mathbf{x}_2$ is the same as the slope of $\mathbf{x}_1$. Similarly, the same argument yields

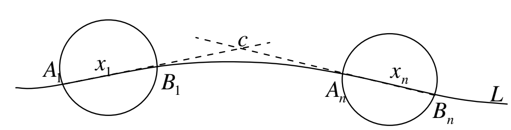
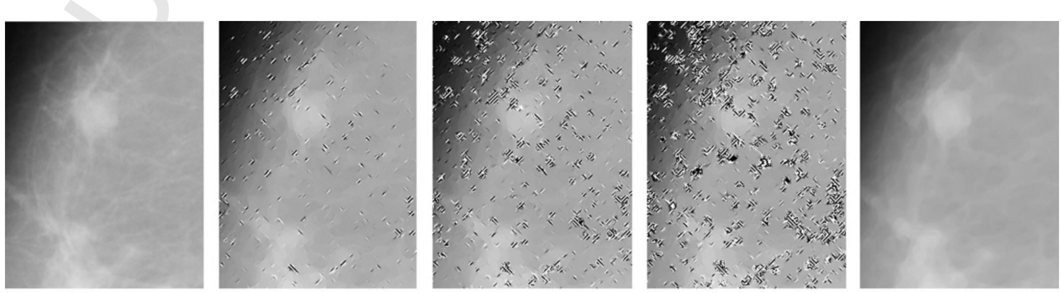


Fig. 4. Illustration of the conflict between the bending edge L and the segments of local points.



a. original image b. iteration=30 c. iteration=60 d. iteration=90 e. iteration=60

Fig. 2. The speckle appear in (b–d) when using Eq. (1), but they are suppressed in (e) when using Eq. (2).

that the adjacent point to $\mathbf{x}_n$, i.e. $\mathbf{x}_{n-1}$, has the same slope as $\mathbf{x}_n$. Obviously, extending the adjacent points to point C, respectively, from $\mathbf{x}_n$ and $\mathbf{x}_1$, one can see that the slope of C is, respectively, equal to the slopes of $\mathbf{x}_n$ and $\mathbf{x}_1$. This means that the two line segments $\overline{A_1B_1}$ and $\overline{A_nB_n}$ have the same slope. Thus, the straight lines $\overline{A_1C}$ and $\overline{B_nC}$ overlap, and $\overline{A_1C}, \overline{B_nC} \subset L$. It is clear that the edge L has to be a straight line. This conclusion conflicts with the initial assumption, and we conclude that L must be a straight line. If the length of L is limited, the continual application of the scheme of Eq. (2) will first smooth out the corners at the endpoints of L, and then erode the center part of L gradually. Finally, this will result in the diffusion of all the structures. From the above analysis, we can state the following proposition.

Proposition 2. *The final steady state solution of the scheme of Eq. (2) falls into one of the following two cases:*

(1) *If the inflection edge becomes a straight line in the neighborhood with an arbitrary radius level around every point that is on the shape boundary, the edge of the shape must be a straight line over the whole image plane;*
(2) *Otherwise, all the features are smoothed out.*

The regular curvature equation usually satisfies the maximum/minimum principle, i.e. the solution does not have a local maximum or minimum at time $t > 0$, and the global extrema are bounded by the initial and the boundary conditions. The boundary conditions usually refer to Neumann boundary conditions ($\partial I/\partial \mathbf{n} = 0$ where $\mathbf{n}$ is the direction of the gradient). It is clear that the global extrema occurs at the initial time, i.e. $I_0(\mathbf{x})$, and the steady state solution is a constant function. Grayson's theorem [4] implies that the shapes or boundaries driven by the curvature flow collapse to a point, and all image information die out. This is the steady state solution of the regular curvature equation. Indeed, the decision rule in Eq. (2) results in the same steady state solution as the one of the regular curvature flow according to Proposition 2.

Another interesting phenomenon is that many continued iterations of the scheme of Eq. (2) with some small radius is roughly similar to one application of the scheme of Eq. (2) with some large radius. Assume that the length of some edge l is greater than the radius, $|l| > r$, no diffusion takes place at the center of l in this case, while the smoothing takes place at the two endpoints of l under the scheme of Eq. (2). It can be noticed that the length of l is shortening through the continued application of the scheme of Eq. (2). When $|l| < r$, the diffusion takes place at the center of l, and the edge l is smoothed out. If the radius is selected so large that the initial edge l satisfies $|l| < r$ at the beginning, it is obvious that the diffusion takes place at every point on l. In this case, the edge l is smoothed out quickly. On that basis, we can formulate the following proposition.

Proposition 3. *Many iterations of the scheme of Eq. (2) with a small radius is roughly equivalent to an evolution with a large radius.*

In Fig. 5(b and c), we illustrate the results of the evolution of Eq. (2) with a *radius* = 1. Fig. 5(d and e) show the result of the evolution of Eq. (2) with a *radius* = 5. We can notice that Fig. 5(d) is similar to Fig. 5(b) despite the difference in the number of iteration and radii. When the iteration numbers are equal, the evolution result with radius = 5 is more blurry than the result with radius = 1 as shown in Fig. 5(b and e). In addition, from Fig. 5(a–c), it can be seen that the continued iteration of this min/max flow results in the erosion of the structure of the object.

3.3. Our modified schemes

It is ideal that the algorithm can stop automatically once some scale noise is removed, and the continued iteration of the min/max flow scheme will not produce further smoothing. Hence, a stricter stopping criterion is required. Although the scheme of Eq. (2) does smooth out all the structures, the min/max switch function is a flexible computational framework, in which many methods and strategies can be combined easily to devise an effective stopping criterion. Indeed, under the min/max flow scheme, we have two choices, one is to modify the velocity term and another is to re-define the decision rules.

3.3.1. Modified scheme 1

First, we present a new scheme by re-defining the decision rules. Consider the heat operator (ΔI) to be applied to the image intensity function. It is known that isotropic smoothing takes place in all directions. In this case, the boundaries of shapes are smeared. Conversely, the inverse heat equation could deblur or enhance an image. The famous example is the shock filter, in which the sign of the Laplacian, $sign(\Delta I)$, is used to determine the evolving direction of the flow. Indeed the change of $sign(\Delta I)$ indicates that the front of the current flow should be on some boundaries. The reverse heat equation would enhance these boundaries. In the curvature evolution equation, the diffusion should take place in a direction orthogonal to the gradient, whereas only the boundaries in the gradient direction need to be enhanced. This is easily fulfilled by replacing the Laplacian operator with the second derivative of the image intensity in the direction of the intensity gradient, $I_{\eta\eta}$, where η is the direction of the gradient. On this basis, the scheme of Eq. (2) can be re-written as,

$$F = \begin{cases} \max(\kappa, 0), & \text{if } \begin{cases} \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) < \text{Threshold}\& \\ I(\mathbf{A}) < \text{Threshold}\& \\ edgef > 0 \end{cases} \\ \min(\kappa, 0), & \text{if } \begin{cases} \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) > \text{Threshold}\& \\ I(\mathbf{A}) > \text{Threshold}\& \\ edgef > 0 \end{cases} \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

a. original image | b. iteration=140, radius=1 | c. iteration=360, radius=1 | d. iteration=80, radius=5 | e. iteration=140, radius=5

Fig. 5. The comparison of the results of evolution of Eq. (2) with different radii. Due to the different radius, many iterations of Eq. (2) with a small radius is roughly equivalent to an evolution with a large radius.

Please cite this article in press as: H. Yu et al., An extension of min/max flow framework, Image Vis. Comput. (2008), doi:10.1016/j.imavis.2008.05.006

where $edgef = \max\{sign((G_\sigma^* I_{\eta\eta})|_{\mathbf{X}}), \mathbf{X} \in \Omega_{3\times3}(\mathbf{A})\} \cdot \min\{sign((G_\sigma^* I_{\eta\eta})|_{\mathbf{X}}), \mathbf{X} \in \Omega_{3\times3}(\mathbf{A})\}$. The term *edgef* in Eq. (3) is namely a zero-crossing detector of the second derivative of the image. When $edgef > 0$, the edges are outside the detection window $\Omega_{3\times3}$, and in this case the diffusion takes place. On the other hand when $edgef \leqslant 0$, the window $\Omega_{3\times3}$ should straddle the edges, and no diffusion takes place. In practice, this kind of zero-crossing method is usually implemented with the Canny edge detector [19]. In this scheme, we firstly apply the Canny detector on a sub-image of size n by n centered around some point A. Then, the window $\Omega_{3\times3}(\mathbf{A})$ around the point A is defined on the output binary image from the Canny detector. Thus, one can appreciate that the *edgef* term in Eq. (3) can be easily implemented.

It can be easily verified that when the front of the current flow is on some boundaries of shapes, the change of the sign of the second derivative halts the motion of the min/max flow. The result is that any further diffusion across the edges is suppressed. In this way, the boundaries of shapes are preserved, and smoothing only takes place inside the region of the object, but not across the boundaries. Hence, the scheme of Eq. (3) can reach a steady state solution. In Fig. 6, we illustrate the evolution results when using the scheme of Eqs. (2) and (3) with *radius* = 1 on a mammographic image. Fig. 6(d) demonstrates that the schemes of Eqs. (2) and (3) reach a steady state solution. The evolution error curve is calculated using, $\mathrm{Error}(t) = \frac{1}{M\times N}\sum_{i=1,j=1}^{M,N}|I_{i,j}^{(t)} - I_{i,j}^{(0)}|$, where M and N are the width and height of the image, respectively. It can be noticed that the scheme of Eq. (3) stops automatically when all textures below the radius are removed, and the continued application of Eq. (3) will not produce further changes. This is in contrast to the results of the scheme of Eq. (2), which does not terminate automatically. Continued evolution of the scheme of Eq. (2) results in the disappearance of the texture and the erosion of the structure.

3.3.2. Modified scheme 2

In this section, we present another scheme by introducing the Gradient Vector Flow (GVF) field into the velocity term. The GVF field was first presented for the active contour model in [11]. It is usually computed as a diffusion of the intensity gradient vectors that enable noise to be suppressed. Since the GVF is estimated directly from the continuous gradient space, and its measurement is contextual and not equivalent to the distance from the closest point. Besides, the GVF provides a bi-directional force field that captures the object boundaries from either side. This provides a reasonable evolving direction for the curvature flow, and also leads the modified scheme to a steady state solution.

First, a Gaussian edge detector is used in the edge mapping function $f(\mathbf{x})$ defined by,

$$f(\mathbf{x}) = 1 - \frac{1}{\sqrt{2\pi}\sigma_E}\exp\left\{-\frac{|\nabla(G_\sigma^* I)(\mathbf{x})|^2}{2\sigma_E^2}\right\},$$

where $\mathbf{x} \in R^2$, G_σ is the Gaussian filter with zero-mean and σ variance, and σ_E determines the width of $f(\mathbf{x})$. In general, σ is independent from σ_E. For convenience, the Gaussian operator G_σ in $f(\mathbf{x})$ is usually omitted. The GVF field $\vec{\mathbf{v}}(\mathbf{x})$ is defined as the equilibrium solution to the following vector diffusion equation,

$$\begin{cases} \vec{\mathbf{v}}(\mathbf{x})_t = \mu\nabla^2\vec{\mathbf{v}}(\mathbf{x}) - f(\mathbf{x})(\vec{\mathbf{v}}(\mathbf{x}) - \nabla f(\mathbf{x}))|\nabla f(\mathbf{x})|^2 \\ \vec{\mathbf{v}}(\mathbf{x}, 0) = \nabla f(\mathbf{x}) \end{cases}$$

where μ is a blending parameter. Fig. 7 illustrates $f(x)$ and $\vec{\mathbf{v}}(x)$ in the 1D case. It can be noted that there are peaks in $f(x)$ corresponding to the edges of the original signal $I(x)$, and the directions of $\vec{\mathbf{v}}(x)$ are changed nearby the edges of $I(x)$, i.e. $\vec{\mathbf{v}}(x)$ points to the edges.

The GVF field contains mainly contextual information and the flow vectors of this field always point to the closest object boundaries. The $\vec{\mathbf{v}}(\mathbf{x})$ is dot-multiplied by the unit gradient vector, $\mathbf{N} = \nabla I/|\nabla I|$, as follows,

$$\vec{\mathbf{v}} \cdot \mathbf{N} = \lambda\nabla|\nabla I| \cdot \mathbf{N} = \lambda\left\langle D^2 I\frac{\nabla I}{|\nabla I|}, \mathbf{N}\right\rangle = \lambda I_{\eta\eta},$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product of two vectors, $D^2 I$ denotes the Hessian of the image intensity I, and $\lambda(|\nabla I|) = \frac{|\nabla I|}{\sqrt{2\pi}}\sigma_E^3\exp\left\{-\frac{|\nabla I|^2}{2\sigma_E^2}\right\} > 0$. It can be noted that the above equation is equal to the second derivative of I in the direction of the intensity gradient, $I_{\eta\eta}$, up to a positive scale λ. Around the boundaries, the sign of $(\vec{\mathbf{v}} \cdot \mathbf{N})$ changes along a normal direction to the boundaries even if the direction of

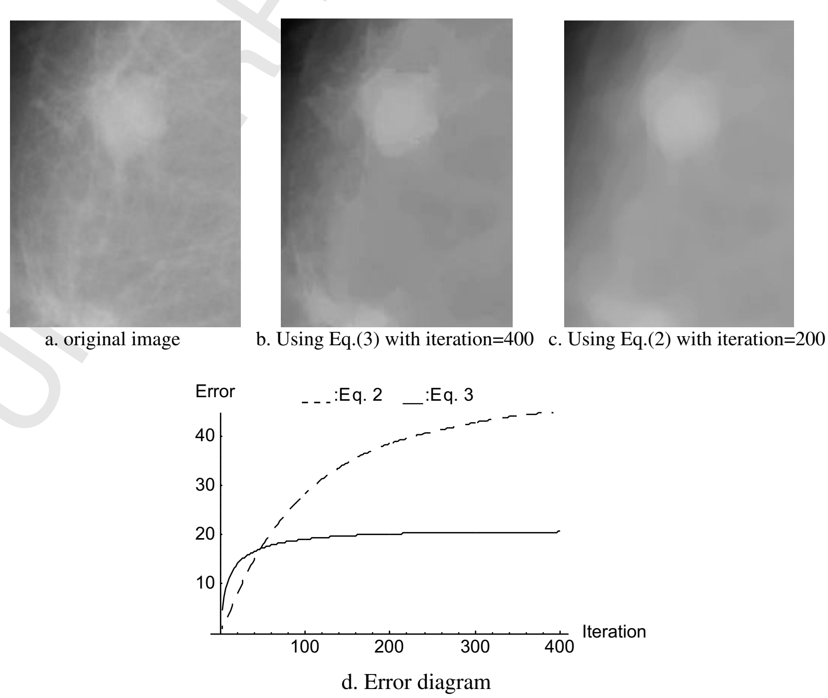


Fig. 6. Comparison of the convergence of Eqs. (2) and (3).

Please cite this article in press as: H. Yu et al., An extension of min/max flow framework, Image Vis. Comput. (2008), doi:10.1016/j.imavis.2008.05.006

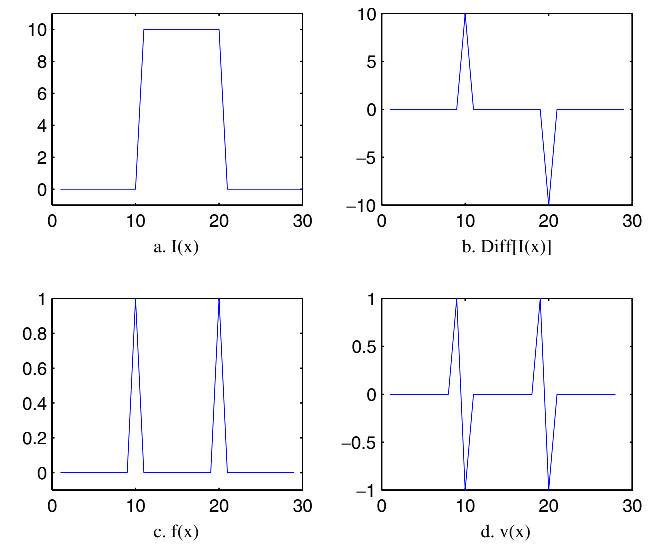


Fig. 7. Illustrations of the edge mapping function and the GVF. Notice that the edge mapping function contains the peaks corresponding to the edges of the original signal as shown in (c), and the GVF can change its direction across the edges as shown in (d). (Herein, $\vec{\mathbf{v}}(x)$ is computed only using one order derivative of $f(x)$).

the gradient $\mathbf{N}$ does not change. This means that the GVF indicates a reasonable evolution direction of the curvature flow, but not the direction of the intensity gradient. Thus, the optimal way to reach the boundaries is to move along the direction of GVF. Given the fact that the propagation of the curvature flow always takes place in the inward normal direction, it can be noted that when the direction of $\vec{\mathbf{v}}(\mathbf{x})$ and the inward normal direction, $-\mathbf{N}$, are identical, the diffusion takes place in the inward normal direction. On the other hand, when they have opposite directions, the diffusion should take place in the outward normal direction $\mathbf{N}$. Because of the noise or spurious edges, the inward normal direction cannot always align with $\vec{\mathbf{v}}(\mathbf{x})$. Hence, the worse case occurs when $\vec{\mathbf{v}}(\mathbf{x})$ is tangent to the normal $\mathbf{N}$. However, the speed function κ in the case of the curvature flow can be modified as follows,

$$\hat{\kappa} = sign(-\vec{\mathbf{v}} \cdot \mathbf{N})|\kappa|.$$

Under the min/max flow framework, the speed function becomes,

$$F = \begin{cases} \max(\hat{\kappa}, 0), & \text{if } \begin{cases} \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) < \text{Threshold} \\ I(\mathbf{A}) < \text{Threshold} \end{cases} \\ \min(\hat{\kappa}, 0), & \text{if } \begin{cases} \text{average}(I(\mathbf{X}), \mathbf{X} \in \Omega(\mathbf{A})) > \text{Threshold} \\ I(\mathbf{A}) > \text{Threshold} \end{cases} \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

From Proposition 2, we know that the curvature flow κ under the min/max flow framework smoothes out all the structure over the whole image. When the curvature flow κ is replaced by $\hat{\kappa}$, it can be noted that the evolving direction of the curvature flow should be determined by $sign(-\vec{\mathbf{v}} \cdot \mathbf{N})$. Consider a small neighborhood of a point on the boundaries of shape Ω. We assume that the GVF and the inward normal direction have the same direction, i.e. $sign(-\vec{\mathbf{v}} \cdot \mathbf{N}) = 1$. If the max flow is selected according to the decision rules of Eq. (4), the convex of the shape in Ω is smoothed. If the curvature flow runs across the boundary of the shape, the GVF and the inward normal direction will have opposite directions in the next iteration, i.e. $sign(-\vec{\mathbf{v}} \cdot \mathbf{N}) = -1$. Even if the max flow is selected according to the decision rules in Eq. (4), the speed function is set to zero, i.e. $F = 0$, and therefore any the further diffusion would be suppressed. So, according to the above analysis, the final steady state image should preserve the essential structures of the shape.

Indeed, the flow under $F = \hat{\kappa}$ is basically the shock filter. Expanding $\hat{\kappa}$ will yield,

$$\hat{\kappa} = -sign\left(\lambda\left\langle D^2 I \frac{\nabla I}{|\nabla I|}, \mathbf{N}\right\rangle\right)|\kappa| = -sign(\lambda I_{\eta\eta})|\kappa|.$$

Its evolution equation is written as $I_t = -sign(\lambda\ I\eta\eta) \cdot |\kappa| \cdot |\nabla I|$. Clearly this is the standard shock filter equation up to a positive scale $|\kappa|$. The scheme of Eq. (4) may be viewed as an implementation of the shock filter under the min/max flow framework. As a matter of fact, the classical shock filter is extremely sensitive to noise. Whereas, the scheme of Eq. (4) performs well on image enhancing and denoising. This is because the original shock filter is an inverse heat equation and therefore sensitive to noise [27], whereas the min/max switch function can effectively suppress the noise enhancement in the scheme of Eq. (4).

The distinct advantage of the GVF fields is to provide a large capture range for the edges. This is in favor of denoising. For comparison, we first illustrate the original shock filter and the scheme of Eq. (4), respectively, on a noisy and blurry image. The original image is the same mammographic image as in Fig. 6(a), which is degraded with a Gaussian noise (zero-mean and 0.1 variance) in Fig. 8(b). Obviously, the original shock filter generates many spurious boundaries as shown in Fig. 8(c). Fig. 8(e) further demonstrates the scheme of Eq. (4) reaching a steady state solution with the evolution error diagram. Thus, the presented scheme of Eq. (4) can effectively remove noise while simultaneously preserving some essential features of the object. The original medical image in Fig. 8(a) is very blurry, and its luminous contrast is very low. In order to illustrate the properties of the scheme Eq. (4), we apply this scheme on a nature scene image, so that its advantages become more evident. In Fig. 9(b), the original water lily image is blurred and degraded with Gaussian noise. We can see that the features of the lily image are enhanced effectively, and the final steady state image can preserve the essential details in Fig. 9(c). However, the final effect of diffusion also relies on the GVF fields. The ideal case is that all the essential shape details should be preserved in the GVF fields.

4. The extended min/max flow framework

The min/max flow framework is a flexible computational framework. In this section, we will extend this framework and incorporate it with the active contour model. Because the boundary leaking problem exists in the standard deformable models, we will demonstrate that the incorporation of this extended framework is ideal to overcome this major shortcoming. We will also illustrate its applications to segmentation and region tracking.

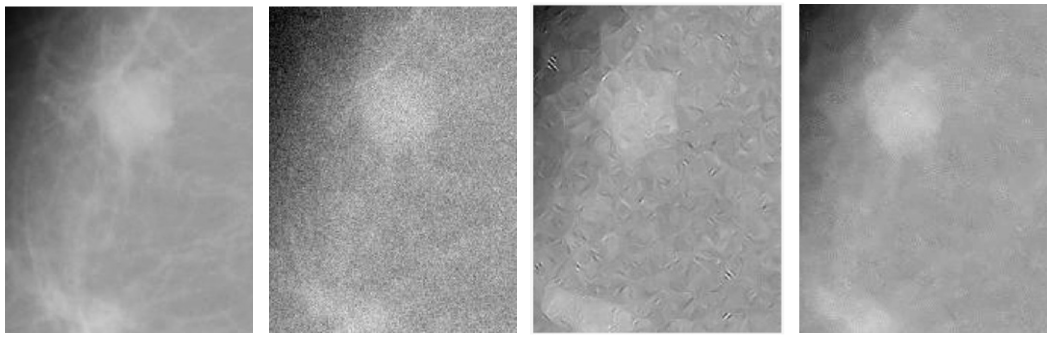
4.1. Extended min/max flow framework

In the context of the general level set equation $\phi_t = F|\nabla\ \phi|$, F is called the speed function, which corresponds to the speed of the front (or evolving curve) in a direction opposite to the normal of the front. The extended min/max flow framework can be written as,

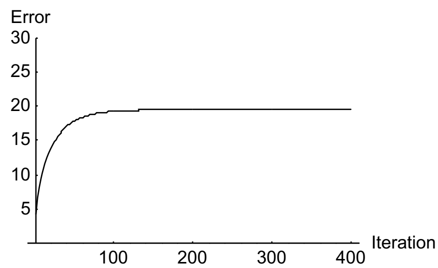
$$F(\mathbf{X}) = \begin{cases} \min(F_c(\mathbf{X}), 0), & \text{satisfyingdecisionrules} \\ \max(F_c(\mathbf{X}), 0), & \text{otherwise} \end{cases}, \qquad (5)$$

where F_c is the speed function of the familiar curvature flow, and the decision rules can be designed specially for various applications.

First, we consider the following two flows, $F = \min(F_c, 0)$ and $F = \max(F_c, 0)$, whose properties are described in Section 2. When

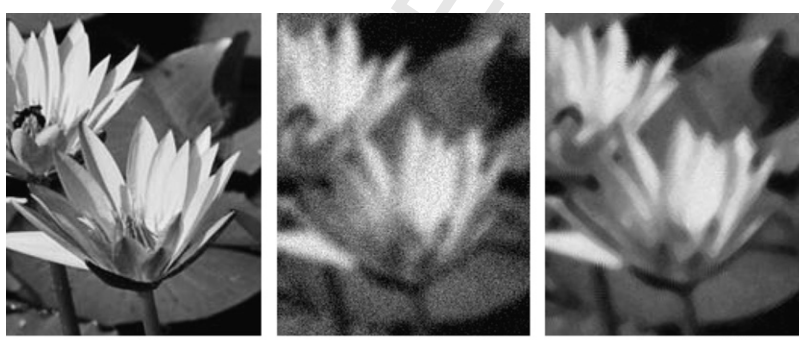


a. Original image b. Gaussian noisy image c. original shock filter-based d. Scheme of (4)



e. Error diagram

Fig. 8. Comparison of evolution results when using Eq. (4) and the original shock filter on the Gaussian noisy image in terms of the number of iterations.



a. original image b. blurry and Gaussian noisy image c. iteration=200

Fig. 9. The distinct water lily image in (a) is blurred and noised with Gaussian in (b). It is then restored by the scheme of Eq. (4) in (c). Notice that the restored image (c) is denoised and deblurred compared to image (b).

the level set function ϕ is chosen, the negative of the signed distance is in the interior, and the positive sign is in the exterior region. The flow under $F = \min(F_c, 0)$ will tend to grow outwards endlessly. Conversely, the flow under $F = \max(F_c, 0)$ will tend to shrink until it collapses to a point. Very roughly speaking, we can think of the choice of the max and min flow as somewhat related to the evolving tendency, i.e. the evolving tendency of the flow under Eq. (5) depends on the choice of the max and min flows, while the evolving tendency of the flow under $F = F_c$ only depends on F_c.

Then, let the flow under $F = F_c$ corresponding to the level set equation $\phi_t = F_c|\nabla\phi|$ be able to reach a trivial steady state solution. Note that the steady state solution is only a local optimal solution of the above equation, and it is not unique. The choice of the flow under Eq. (5) is either F_c or zero. Let us now consider the flow under the extended framework of Eq. (5) and the flow under $F = F_c$ together. If the choice of the flow resulting from the speed function of Eq. (5) and the flow under $F = F_c$ are identical at all time, it is clear that a steady state solution of Eq. (5) corresponds to one of the flows under $F = F_c$. On the other hand, the worst case occurs when the choice resulting from the speed function in Eq. (5) and the flow under $F = F_c$ have opposite evolving tendencies, i.e. the speed is always set to zero, $F = 0$. In this case, no propagation takes place under the scheme of Eq. (5). In addition, if the min/max flow happens to stay at some state that is a steady state for the flow under $F = F_c$, it is clear in this case that $F_c = 0$. Then, regardless of the choice of the max or min flow, in either case, there will be no propagation that takes place.

When the flow under the extended framework of Eq. (5) and the flow under $F = F_c$ have the same evolving tendency on some local parts of the evolving curve, these parts will tend to a steady state just as they are driven by the flow under $F = F_c$. When the opposite evolving tendency appears on other parts of the evolving curve, then there is no propagation that takes place. Hence, propagation takes place only if the choice of the flow under Eq. (5) and the flow under $F = F_c$ are identical. It is similar to the case where the front of propagation is driven by the flow under $F = F_c$. The final steady state solution of the scheme of Eq. (5) should be bounded by all steady state solutions that the flow under $F = F_c$ can reach. From the above analysis, we can state the following proposition.

Please cite this article in press as: H. Yu et al., An extension of min/max flow framework, Image Vis. Comput. (2008), doi:10.1016/j.imavis.2008.05.006

Proposition 4. *If the flow under $F = F_c$ can reach a steady state, then the flow under the extended min/max flow framework will also be able to reach a steady state.*

It is clear that the min/max framework only helps us to select an appropriate solution from the steady state solution set of the evolving flow under $F = F_c$. However, this extended framework is an open computational framework, under which many constraint conditions can be incorporated in the speed function to suppress boundary leaking. In the following sections, we will apply this extended framework to image segmentation and region tracking in order to suppress the boundary leaking problem.

4.2. Boundary leaking

The boundary leaking problem was first observed in the case of the segmentation of CT medical image in [12]. In fact, it is an intrinsic fault of the original geodesic active contour model presented in [20]. The geodesic active model is usually written as,

$$\mathbf{C}_t = g(I)\kappa_c\mathbf{N}_c - (\nabla g \cdot \mathbf{N}_c)\mathbf{N}_c,$$

where κ_c is the Euclidean curvature of the evolving curve $\mathbf{C}(t)$, $\mathbf{N}_c$ is the unit normal inward of $\mathbf{C}(t)$, $g(I)$ is usually defined as a monotonically decreasing function of the intensity gradient $|\nabla I|$, which attracts the evolving curve towards the object boundary. Using the level set framework, we can obtain its level set representation as follows,

$$\phi_t = g(I)\kappa_c|\nabla\phi| + \nabla g \cdot \nabla\phi \tag{6}$$

with the initial condition $\phi(0,C) = \phi_0(C)$, and its speed function $F = g(I)\kappa_c - \nabla g \cdot \mathbf{N}_c$. It is easy to see that the curvature term (the first term) in the speed function is the known Euclidean heat flow, which is used for curve smoothing and evolution. In practice, if only the dynamic balance is reached between the curvature term and the second term in the speed function, the scheme of Eq. (6) can reach a steady state solution. Expanding further the speed function F will yield,

$$F = g(I)\kappa_c - g'\left\langle D^2 I\frac{\nabla I}{|\nabla I|}, \mathbf{N}_c\right\rangle$$

Let us assume that the evolving curve has converged to the real boundary of the object. The unit vector of intensity gradient and the normal of $C(t)$ are identical in this case, $-\mathbf{N}_c = \nabla I/|\nabla I|$. Thus, the second term should be the second derivative of I in the direction of the intensity gradient up to a derivative factor $g'(I)$, and is equal to zero, i.e. $\left\langle D^2 I\frac{\nabla I}{|\nabla I|}, \mathbf{N}_c\right\rangle = I_{\eta\eta} = 0$. Because the function $g(I) = minimum \neq 0$ in this case, it is difficult to reach a dynamic balance state at the real boundary. Thus, this scheme is prone to boundary leaking.

The boundary leaking problem does not only appear in image segmentation, but also appears in region tracking applications. Usually, most deformable models can be used to extract the contours of moving objects in the tracking region, but the input frames need to be enhanced in advance. Region tracking is usually based on the observed inter-frame intensity difference model, $d(x,y) = I(x,y,t) - I(x,y,t-1)$, where $I(x,y,t)$, $I(x,y,t-1)$ are the current and previous frame intensity functions. The probability density function of $d(x,y)$ is usually modeled as a mixture of two Gaussian (or Laplacians) distributions, which are both zero-mean and correspond to the background area and the moving objects area, respectively.

In [21,22], the detection and tracking of moving objects in the image sequences were formulated in a variational framework. Motion detection estimates only the moving area between the two successive frames. It does not detect the input frame sequences directly, but rather detects a new generated frame, which is described as,

$$I_D(\mathbf{X}) = \max_{\mathbf{Y}\in\Omega(\mathbf{X})}\left\{\frac{pd|_{bg}(d(\mathbf{X}))\cdot pd|_{obj}(d(\mathbf{Y})) + pd|_{obj}(d(\mathbf{X}))\cdot pd|_{bg}(d(\mathbf{Y}))}{pd|_{bg}(d(\mathbf{X}))\cdot pd|_{bg}(d(\mathbf{Y})) + pd|_{obj}(d(\mathbf{X}))\cdot pd|_{obj}(d(\mathbf{Y}))}\right\},$$

where $\Omega(\mathbf{X})$ denotes the neighborhood of pixel $\mathbf{X} \in R^2$, and $pd|_{bg}(,pd|_{obj})$ are the probability density function of the observed inter-frame difference $d(\mathbf{X})$ under the background area (or object area) hypothesis (for more details, refer to [21,22]). In $I_D(\mathbf{X})$, the moving area has been enhanced. This can be observed in Fig. 12(b). The geodesic active contour model based edge detection operation can then effectively be used for the motion detection. This is then followed by a tracking module. Unlike motion detection, the tracking part detects the boundaries of the moving object on the original input frames directly.

The detection and tracking problems were described in [21] as an energy minimization problem. The Euclidean curve evolution equation is described as,

$$\begin{aligned}\mathbf{C}_t = {} & \gamma(g(I_D)\kappa_c - \nabla g(I_D)\cdot\mathbf{N}_c)\mathbf{N}_c + (1-\gamma)(g(|\nabla I|)\kappa_c \\ & - \nabla g(|\nabla I|)\cdot\mathbf{N}_c)\mathbf{N}_c, \quad \gamma \in [0,1]\end{aligned}$$

Its level set representation can be written as,

$$\begin{aligned}\phi_t = {} & \gamma\left(g(I_D)\kappa_c + \nabla g(I_D)\cdot\frac{\nabla\phi}{|\nabla\phi|}\right)|\nabla\phi| + (1-\gamma)(g(|\nabla I|)\kappa_c \\ & + \nabla g(|\nabla I|)\cdot\frac{\nabla\phi}{|\nabla\phi|})|\nabla\phi|.\end{aligned} \tag{7}$$

It is clear that both the motion detection and tracking parts adopt the geodesic active contour model, whose speed functions have the common form as reported in Eq. (6). Thus, the boundary leaking problem also appears in the scheme of Eq. (7) as well.

Through the above analysis, one can see that the boundary leaking problem is an intrinsic fault of the geodesic active contour model. Similarly, it also appears in the parametric active models. By way of adding a conservative (non-conservative) external force, this problem is at most alleviated but cannot be surmounted. The extended min/max framework can help us to select an appropriate solution from the trivial steady state solution set of the curvature flow. This solution may not be the optimal solution but is a reasonable approximation. This is because it is possible to suppress boundary leaking when the wrong evolving tendency of the flow can be corrected.

In the following sections, the schemes of Eqs. (6) and (7) are redefined under the extended min/max flow framework, and the decision rules are modeled as probability density functions.

4.3. Image segmentation under the extended min/max flow framework

Our proposed algorithm aims at the partial volume estimation [23], which is essentially a boundary leaking problem. Usually, these kinds of problems indicate that the risk of misclassification for adjacent regions is too big, and in some extreme cases where the two distributions have the same mean but different variances, the classification error is intolerable. In order to overcome these problems, some further texture features are extracted and added to the feature vector. In this paper, for convenience, we consider Gaussian distributions. We also assume that the number of classes is known a priori so as to estimate the parameters of the statistical model.

In order to efficiently deal with the finite mixture model, we choose some clique types as shown in Fig. 10. Besides image intensity is also used to construct the pixel feature vector. A similar idea can be found in [24]. The feature vector $\mathbf{V}(\mathbf{X}) = (v_1(\mathbf{X}), \ldots, v_m(\mathbf{X}), v_{m+1}(\mathbf{X}))^{\mathrm{T}}, \mathbf{X} \in R^2$ is defined by,

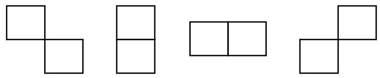


Fig. 10. The four clique types associated to a second order model.

$$\begin{cases} v_i(\mathbf{X}) = \sum\limits_{c \in C_i} \Delta_c(\mathbf{X}), & i \leqslant m \\ v_{m+1}(\mathbf{X}) = \frac{1}{w \times w} \sum\limits_{X \in W} I(\mathbf{X}), & i = m+1 \end{cases}$$

where C_i is the set of all cliques of type i in a given window $W(w \times w)$, $\Delta_c(\mathbf{X}) = -1$ if $I(\mathbf{X}) = I(\mathbf{X}')$ and $\Delta_c(\mathbf{X}) = 1$ otherwise. It is easy to see that the first m elements are texture features while the $(m+1)$th is an intensity feature. In our algorithm, the length of the feature vector is 5. These features can then be applied to both textured and non-textured images.

Suppose that the feature vectors satisfy a Gaussian distribution, i.e. $\mathbf{V}(\mathbf{X}) \sim N(\mathbf{m},\Sigma)$. Once these feature vectors have been extracted, the EM algorithm is applied to estimate the parameters of each class in the image. Since the elements in $\mathbf{V}(\mathbf{X})$ are not independent of each other, the probability density function $pd_i(\mathbf{V}(\mathbf{X}))$ should be of a higher dimension Gaussian type for each class $R_i, R = \bigcup_i R_i$. In order to estimate the probability of each pixel robustly, one can sample a neighbor window around each pixel. For the partial mixture case in a given window, the above probability density function can be replaced by the joint probability density function for each class,

$$\log pd_i(\mathbf{V}(\mathbf{X})|\mathbf{m}_i, \Sigma_i) = \sum_{\mathbf{Y} \in W(\mathbf{X})} \alpha_i \log pd_i(\mathbf{V}(\mathbf{Y})|\mathbf{m}_i, \Sigma_i),$$

where $\alpha_i = \|R_i \cap W\|/(w \times w)$. This makes the mixture distribution of $\mathbf{X}$, $\Pr(\mathbf{V}(\mathbf{X})) = \prod_i pd_i(\mathbf{V}(\mathbf{X}) \mid \mathbf{m}_i, \Sigma_i)$, depend on the location $\mathbf{X} = (x,y)^{\mathrm{T}} \in R^2$. Because $\mathbf{V}(\mathbf{X}) \sim N(\mathbf{m},\Sigma)$ for a given window $W(\mathbf{X})$, its sample mean $\overline{\mathbf{V}}(\mathbf{X})$ and covariance matrix $\overline{\Sigma}(\mathbf{X})$ should also follow a Gaussian distribution. In [25], the above joint probability density function for each class was re-written as,

$$\log pd_i(\mathbf{V}(\mathbf{X})|\mathbf{m}_i, \Sigma_i) = -\frac{1}{2}\Big[(\overline{\mathbf{V}}(\mathbf{X}) - \mathbf{m}_i)^{\mathrm{T}} \Sigma_i^{-1} (\overline{\mathbf{V}}(\mathbf{X}) - \mathbf{m}_i) + tr(\Sigma_i^{-1}\overline{\Sigma}(\mathbf{X})) + \log(2\pi \det(\Sigma_i))\Big].$$

It is clear that the above equation can detect two regions with the same mean but with different variances.

Consequently, a Bayesian classification method is applied to decide to which class each pixel should belong using the following likelihood function,

$$\Pr(\mathbf{X}|\mathbf{m}_i, \Sigma_i) = \frac{\Pr(\mathbf{m}_i, \Sigma_i) \cdot pd_i(\mathbf{V}(\mathbf{X})|\mathbf{m}_i, \Sigma_i)}{\sum_i \Pr(\mathbf{m}_i, \Sigma_i) \cdot pd_i(\mathbf{V}(\mathbf{X})|\mathbf{m}_i, \Sigma_i)},$$

where the prior probability $\Pr(\mathbf{m}_i, \Sigma_i)$ can be estimated using the EM algorithm. The Bayesian classification result is the class number,

$$L(\mathbf{X}) = \arg\max_i(\Pr(\mathbf{X}|\mathbf{m}_i, \Sigma_i)).$$

This is because the inside of the evolving curve is viewed as a single region R_{L_0}, the decision rules are only used to decide on whether the Bayesian classification result $L(\mathbf{X})$ is equal to the class number L_0. Under the min/max flow framework, the curve evolution equation can be re-written as,

$$\phi_t = F|\nabla\phi|, \tag{8}$$

where

$$F = \begin{cases} \min\{(g(|\nabla I|)\kappa_c - \nabla g \cdot \mathbf{N}_c), 0\}, & L(\mathbf{X}) = L_0 \\ \max\{(g(|\nabla I|)\kappa_c - \nabla g \cdot \mathbf{N}_c), 0\}, & \text{otherwise} \end{cases}.$$

Compared with the scheme of Eq. (6), the scheme of Eq. (8) increases the capture range, and selects a reasonable evolution direction for each iteration step. This is due to the fact that the Bayesian decision can control the evolution tendency of the evolving curve under the extended min/max flow framework.

4.4. Region tracking under extended min/max flow framework

The scheme of Eq. (7) consists of two parts. The first deals with motion detection, and the other covers the tracking part. First, let us consider the motion detection equation. The crucial step is to decide on whether a given pixel belongs to a moving area. Since the inter-frame difference probability density function is modeled as a mixture of two Gaussian (or Laplacian) distributions. The probability density function can be written as,

$$pd(d(\mathbf{X})) = P|_{bg} \cdot pd|_{bg}(d(\mathbf{X})) + P|_{obj} \cdot pd|_{obj}(d(\mathbf{X})),$$

where $P|_{bg}(,P|_{obj})$ is a priori probability under the background (, object) area case. A decision rule for the motion detection equation can be easily designed according to the above equation. Under the extended min/max flow framework, the motion detection equation can be re-written as,

$$\phi_t = F_D|\nabla\phi|,$$

where

$$\begin{cases} F_D = \min\left(g(I_D)\kappa_c + \nabla g(I_D) \cdot \frac{\nabla\phi}{|\nabla\phi|}, 0\right), & P|_{bg} \cdot pd|_{bg}(d(\mathbf{X})) < P|_{obj} \cdot pd|_{obj}(d(\mathbf{X})) \\ F_D = \max\left(g(I_D)\kappa_c + \nabla g(I_D) \cdot \frac{\nabla\phi}{|\nabla\phi|}, 0\right), & \text{otherwise} \end{cases}.$$

Comparing with Eq. (8), one can note that the image segmentation approach is applied to the enhancement frame $\{I_D\}$ for motion detection. In this case, because there are only object and background areas in frame $\{I_D\}$, the class number is two. The feature is the scale $d(\mathbf{X})$, which follows a normal distribution $d(\mathbf{X}) \sim N(0,\sigma^2)$. Similarly to the image segmentation case, we sample a neighborhood window around point X. The joint probability density function for each class can be re-written as,

$$\log pd_i(d(\mathbf{X})|\sigma_i^2) = -\frac{1}{2}\left[\log(2\pi\sigma_i^2) + \frac{\bar{d}^2(\mathbf{X})}{\sigma_i^2} + \frac{\bar{\sigma}^2(\mathbf{X})}{\sigma_i^2}\right], \quad i = 1, 2,$$

where $\bar{d}(\mathbf{X})$ and $\bar{\sigma}^2(\mathbf{X})$ are the window mean and variance, respectively. The Bayesian decision can conveniently be obtained in the context of the following likelihood function,

$$\Pr(\mathbf{X}|\sigma_i^2) = \frac{\Pr(\sigma_i^2) \cdot pd_i(d(\mathbf{X})|\sigma_i^2)}{\sum\limits_i \Pr(\sigma_i^2) \cdot pd_i(d(\mathbf{X})|\sigma_i^2)}, \quad i = 1, 2.$$

So, the speed function can be re-defined as,

$$\begin{cases} F_D = \min\left(g(I_D)\kappa_c + \nabla g(I_D) \cdot \frac{\nabla\phi}{|\nabla\phi|}, 0\right), & \Pr(\mathbf{X}|\sigma_{bk}^2) < \Pr(\mathbf{X}|\sigma_{obj}^2) \\ F_D = \max\left(g(I_D)\kappa_c + \nabla g(I_D) \cdot \frac{\nabla\phi}{|\nabla\phi|}, 0\right), & \text{otherwise} \end{cases}. \tag{9}$$

For the tracking part equation, the crucial step is to decide on the evolving direction of the evolution curve (i.e. the direction of the object motion). This is a challenge because the texture background or the edges in the original input frame could often change the evolving direction unpredictably. In [26], it was shown that the probability of each pixel belonging to the inside and outside of the evolving curve could be approximated by the infimum of the inter-frame intensity difference. The estimate of infimum can be used in the decision rules for the tracking part equation under the extended min/max flow framework. The tracking equation part can be re-written as,

$$\phi_t = F_T|\nabla\phi|, \tag{10}$$

where

$$\begin{cases} F_T = \min\left(g(|\nabla I|)\kappa_c + \nabla g(|\nabla I|) \cdot \frac{\nabla\phi}{|\nabla\phi|}, 0\right), & V_{\mathrm{in}} < V_{\mathrm{out}} \\ F_T = \max\left(g(|\nabla I|)\kappa_c + \nabla g(|\nabla I|) \cdot \frac{\nabla\phi}{|\nabla\phi|}, 0\right), & \text{otherwise} \end{cases}$$

and

$$\begin{cases} V_{\text{in}} = \inf\limits_{\{Z:|Z|\leqslant\delta, X+Z\in\Omega\}} (I(\mathbf{X},t) - I(\mathbf{X}+\mathbf{Z},t-1))^2 \\ V_{\text{out}} = \inf\limits_{\{Z:|Z|\leqslant\delta, X+Z\in\bar{\Omega}\}} (I(\mathbf{X},t) - I(\mathbf{X}+\mathbf{Z},t-1))^2 \end{cases}.$$

The parameter δ determines the maximum range of motion. Therefore, it should be set to different values for the different image sequences, respectively.

According to Eqs. (9) and (10), the scheme of Eq. (7) is re-defined as,

$$\phi_t = (F_D + F_T) \cdot |\nabla\phi|. \tag{11}$$

Although the speed function of the motion detection and tracking part have the same form as in image segmentation, they represent two different problems. Each one can be solved using its corresponding decision rule. This makes the speed function regularizing and flexible. Indeed, the extended min/max flow framework is only an open computational framework. Under this framework, we can only consider the intrinsic characteristics of the evolving curve to design the speed function, while leave the "stopping criterion" to the decision rules.

4.5. Experiments and analysis

4.5.1. Image segmentation

Experiments of image segmentation were performed on a grey slice image to segment a leg bone. From the original slice image in Fig. 11(a), it can be observed that the boundary of the bone is blurry, but the texture features in the different regions are distinct. This warranted the use of the texture features with our algorithm. In our experiments, our algorithm is used to deal with the partial volume estimation. An initial segmentation was provided. It can usually be obtained using the K-means method (see Fig. 11(b)). For comparison, the image segmentation is implemented using the schemes of Eqs. (6) and (8), respectively. The function $g(|\nabla I|)$ in the evolution equations is defined as $g(|\nabla I|) = 1/(1 + |\nabla I|^k), k = 1$ or 2.

The segmented image obtained using Eq. (6) is shown in Fig. 11(c). Since some parts of the initial evolving curve have run across the edges of the leg bone while others have not reached the boundary, the obtained evolving curve using Eq. (6) was only able to reach some local spurious edges but not the desired boundary of the leg bone at steady state. One can see that the flow under Eq. (6) leaks through at some locations outside of the desired boundary after the continued application of the scheme of Eq. (6). When the segmentation is driven by Eq. (8), the evolving curve tends towards the boundaries of the leg bone gradually, and nicely converges after a number of iterations. The result of the segmented image using Eq. (8) is shown in Fig. 11(d).

4.5.2. Region tracking

Experiments with region tracking were carried out on an image sequence with a 320 × 240 resolution. The two methods driven by the schemes of Eqs. (7) and (11) are compared from the following three aspects, the motion detection part, the tracking part and the combination of these two parts.

4.5.3. Motion detection

For comparison, we implemented the detection of the moving area using the motion detection part of Eq. (7) and the scheme of Eq. (9). In this experiment, the Gaussian model was adopted. The unknown parameters were estimated using the EM algorithm for creating the enhancement frame $\{I_D\}$ shown in Fig. 12(b). It can be seen that the moving area is appropriately separated from the background. We performed motion detection using the schemes of Eqs. (7) and (9), respectively, on the new frame $\{I_D\}$ until a steady state solution was reached. The results are shown in Fig. 12(c and d). It can be noticed that there is no distinct boundary leaking in Fig. 12(c and d). This can be explained as follows. Both the motion detection part of Eq. (7) and the scheme of Eq. (9) can easily drive the evolving curve to the desired boundary. Because the background is zero-valued in the enhancement frame $\{I_D\}$ of Fig. 12(b), while the motion area is non-zero-valued,

4.5.4. Tracking part

For comparison, the contour of the moving object is tracked through the implementation of the tracking part in Eq. (7) and the scheme of Eq. (10), respectively. The former is only based on the current frame $\{I_n\}$, but the latter is based on both the current frame $\{I_n\}$ and the previous frame $\{I_{n-1}\}$. The evolving curves in Fig. 12(c and d) are defined as the initial zero-level set for the tracking part. The results are shown in Fig. 12(e and f). It can be noticed that the evolving curve using the scheme of Eq. (10) can converge to the boundary of the moving object, while the evolving curve using the tracking part of Eq. (7) gets across the desired boundary.

4.5.5. Combined equation

In this experiment, we illustrate the tracking algorithm via the implementation of Eq. (11) on a real image sequence. For convenience, the initial contour is manually outlined in the first frame 0 of the sequence, and then the contour is tracked from frame 0 to 1, then from frame 1 to frame 2, and so on up to the last frame in the sequence. The total number of frames is 30. The initial unknown parameters in the Gaussian models can be estimated accurately using the contour area in the first frame, and then these parameters could be estimated repeatedly using the known contour area in the previous frame. The results are shown in Fig. 13.

In the first 19 frames, the inter-frame motion is small. In this case, the motion detection part in Eq. (11) could easily locate the motion area, i.e. the evolving curve is very close to the boundaries of the moving object. Therefore, the tracking part in Eq. (11) can refine this evolving curve to ensure converge to the contour of the moving object in a small capture range. The parameter δ is set to a small value.

After frame 20, the moving object becomes faster and the image speed is of about 10 pixels per frame. Although the motion detection part in Eq. (11) can detect the motion area, the detected motion area is larger than the object area. Many background edges

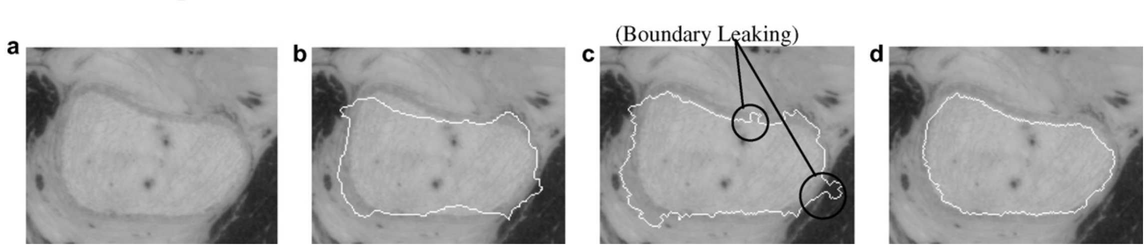


Fig. 11. Image segmentation using the schemes of Eqs. (6) and (8). (a) original image; (b) K-means segmentation; (c) segmentation using Eq. (6); (d) segmentation using Eq. (8). Notice that the scheme of Eq. (8) gives better results and is immune of the boundary leaking problem.

Please cite this article in press as: H. Yu et al., An extension of min/max flow framework, Image Vis. Comput. (2008), doi:10.1016/j.imavis.2008.05.006

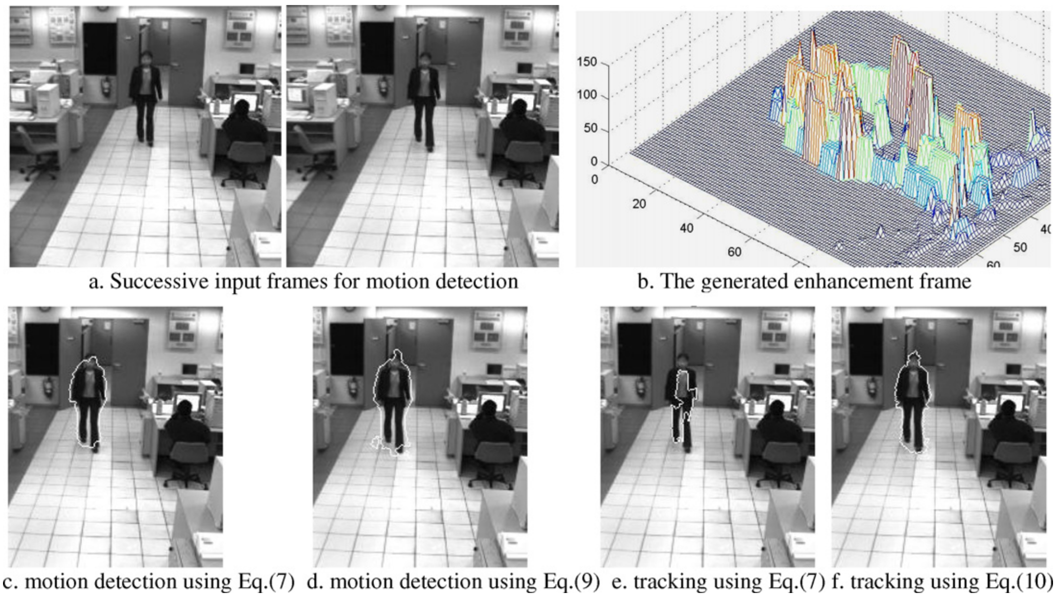


Fig. 12. Comparison of motion detection and tracking part, respectively, using the original scheme of Eq. (7) and our proposed scheme of Eqs. (9) (10). Notice that our proposed scheme gives accurate tracking compared to the original scheme.

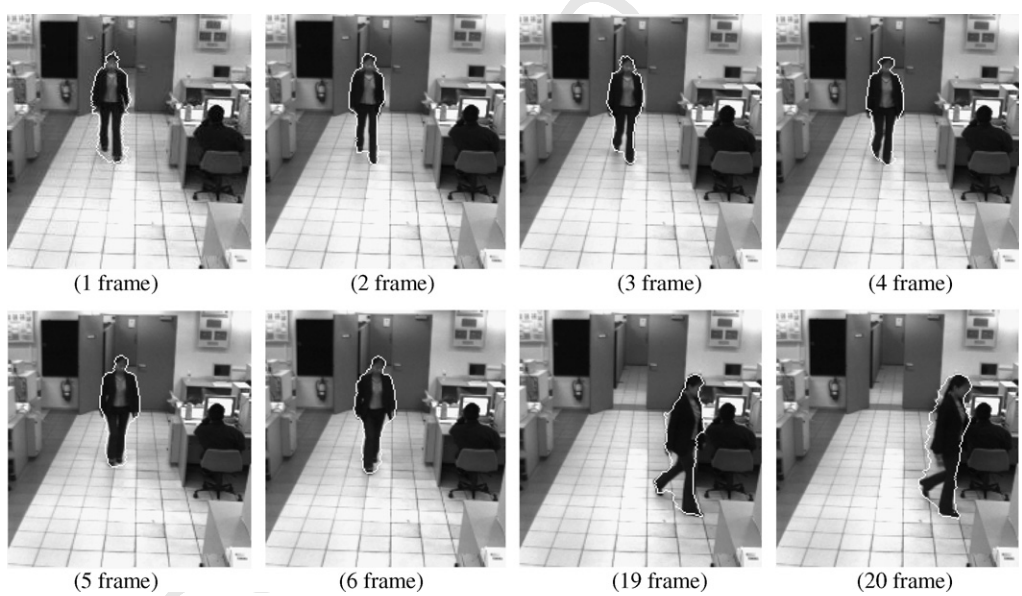


Fig. 13. Region tracking using Eq. (11). In the first six frames, the object is moving slowly, while the object is moving quickly after 19th frame. This shows that the proposed scheme is suitable for slowly moving objects.

or textures appear in this motion area. Hence, the motion detection part did not help in the detection of the object boundaries. It is difficult to track the boundaries of the moving object using the tracking part of Eq. (11) around this complicated area. In frame 20, it can be noticed that the evolving curve converges at the boundaries of the other shapes but not the moving object. Although we tried to carefully adjust the parameter δ to a large value, the contour of the moving object still remained deformed.

5. Conclusions

In this paper, the min/max flow scheme for image enhancement and denoising is revised. The novelty consists in three parts. The first is to analyze the reason behind the speckle generation and the modification of the original scheme. The second is to point out that the continued application of this scheme cannot result in the adaptive stopping of the curvature flow. On that basis, we presented two modified schemes through the introduction of the GVF field and the zero-crossing detector so as to control the smoothing effect. The third contribution is the extension of the min/max flow scheme to deal with the boundary leaking problem. The boundary leaking problem is indeed an intrinsic deficiency of the active contour model. Under the familiar geodesic active contour scheme, none of the existing approaches is able to overcome this shortcoming. Whereas, this proposed min/max flow framework provides an effective way to approximate the optimal solution.

It could be noticed from our experimental results on medical images for enhancement and denoising that the edge contrast is not sufficient under the modified min/max flow schemes. Thus,

IMAVIS 2724
10 June 2008 Disk Used
ARTICLE IN PRESS
No. of Pages 12, Model 5G

we will aim to introduce the inverse diffusion equation in future work. In addition, our experimental results of regions tracking indicate that the extended min/max flow framework only helps us to select an appropriate solution but not the optimal solution. In future work, under this extended framework, our aim will be to develop more robust decision rules.

References

[1] P. Perona, J. Malik, Scale–space and edge detection using anisotropic diffusion, in: IEEE Transactions on PAMI, vol. PAMI-12, no. 7, 1990, pp. 629–639.

[2] S. Osher, L. Rudin, Feature-oriented image enhancement using shock filter, SIAM J. Numer. Anal. 27 (1990) 919–940.

[3] L. Alvarez, F. Guichard, P.L. Lions, J.M. Morel, Axioms and fundamental equations of image processing, Arch. Ration. Mech. Anal. 123 (3) (1993) 199–257.

[4] M. Grayson, Shortening embedded curves, Ann. Math. 129 (1989) 71–111.

[5] R. Malladi, J.A. Sethian, Image Processing: flows under Min/Max curvature and mean curvature, Graph. Image Process. 58 (2) (1996) 127–141.

[6] R. Malladi, J.A. Sethian, A unified approach to noise removal, image enhancement, and shape recovery, IEEE Trans. Image Process. 5 (11) (1996) 1554–1568.

[7] L. Alvarez, P.-L. Lions, J.-M. Morel, Image selective smoothing and edge detection by nonlinear diffusion (II), SIAM, J. Numer. Anal. 29 (3) (1992) 845–866.

[8] T. Chan, A. Marquina, P. Mulet, High-order total variation-based image restoration, SIAM J. Sci. Comput. 22 (2) (2000) 503–516.

[9] M. Lysaker, S. Osher, X.C. Tai, Noise removal using smoothed normals and surface fitting, IEEE Trans. Image Process. 13 (10) (2004).

[10] G. Gilboa, N.A. Sochen, Y.Y. Zeevi, Forward-and-backward diffusion processes for adaptive image enhancement and denoising, IEEE Trans. Image Process. 11 (7) (2002) 689–703.

[11] C. Xu, J. Prince, Snakes, shapes, and gradient vector flow, IEEE Trans. Image Process. 7 (3) (1998) 359–369.

[12] K. Siddiqi, Y. Lauziere, A. Tannenbaum, et al., Area and length minimizing flows for shape segmentation, IEEE Trans. Image Process. 7 (3) (1998) 433–443.

[13] N. Paragios, O. Mellina-Gottardo, V. Ramesh, Gradient vector flow fast geodesic active contours, Proceedings of IEEE International Conference on Computer Vision, vol. 1, Vancouver, Canada, 2001, pp. 67–73.

[14] B. Sumengen, B.S. Manjunath, C. Kenney, Image segmentation using curve evolution and flow fields, in: IEEE International Conference on Image Processing, vol. 1, Rochester, NY, USA, 2002, pp. 105–108.

[15] C. Lu, Y. Cao, D. Mumford, Surface evolution under curvature flows, J. Vis. Commun. Image Representation 13 (1/2) (2002) 65–81.

[16] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001).

[17] T.F. Chan, L.A. Vese, A multiphase level set framework for image segmentation using the Mumford and Shah model, Int. J. Comput. Vis. 50 (3) (2002) 271–293.

[18] R. Kimmel, A.M. Bruckstein, Regularized Laplacian zero crossings as optimal edge integrators, Int. J. Comput. Vis. 53 (3) (2003) 225–243.

[19] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell. 8 (1986) 679–698.

[20] V. Caselles, R. Kimmel, G. Sapiro, Geodesic active contours, Int. J. Comput. Vis. 22 (1997) 61–79.

[21] N. Paragios, R. Deriche, Geodesic active contours and level sets for the detection and tracking of moving objects, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 266–280.

[22] N. Paragios, G. Tziritas, Detection and location of moving objects using deterministic relaxation algorithms, in: Proceedings of IAPR International Conference on Pattern Recognition, Vienna, Austria, 1996, pp. 201–205.

[23] D.H. Laidlaw, K.W. Fleischer, A.H. Barr, Partial-volume Bayesian classification of material mixtures in MR volume data using voxel histograms, IEEE Trans. Med. Imaging 17 (1) (1998) 74–86.

[24] M. Roula, A. Bouridane, A. Amira, et al., A novel technique for unsupervised texture segmentation, in: Proceedings of IEEE International Conference on Image Processing, vol. 1, Thessaloniki, Greece, 2001, pp.58–61.

[25] S.C. Zhu, A. Yuille, Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 18 (9) (1996) 884–900.

[26] A. Mansouri, Region tracking via level set PDE's without motion computation, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 947–961.

[27] G. Gilboa, N.A. Sochen, Y.Y. Zeevi, Regularized shock filters and complex diffusion, in: A. Heyden, et al. (Eds.), Proceedings of Europe Conference on Computer Vision 2002, LNCS 2350, Springer-Verlag, Berlin, Heidelberg, pp. 399–413.

Please cite this article in press as: H. Yu et al., An extension of min/max flow framework, Image Vis. Comput. (2008), doi:10.1016/j.imavis.2008.05.006