Vessel Boundary Tracking for Intravital Microscopy Via Multiscale Gradient Vector Flow Snakes

Jinshan Tang, *Senior Member, IEEE* and Scott T. Acton*, *Senior Member, IEEE*

***Abstract*—Due to movement of the specimen, vasodilation, and intense clutter, the intravital location of a vessel boundary from video microscopy is a difficult but necessary task in analyzing the mechanics of inflammation and the structure of the microvasculature. This paper details an active contour model for vessel boundary detection and tracking. In developing the method, two innovations are introduced. First, the B-spline model is combined with the gradient vector flow (GVF) external force. Second, a multiscale gradient vector flow (MSGVF) is employed to elude clutter and to reliably localize the vessel boundaries. Using synthetic experiments and video microscopy obtained via transillumination of the mouse cremaster muscle, we demonstrate that the MSGVF approach is superior to the fixed-scale GVF approach in terms of boundary localization. In each experiment, the fixed scale approach yielded at least a 50% increase in root mean squared error over the multiscale approach. In addition to delineating the vessel boundary so that cells can be detected and tracked, we demonstrate the boundary location technique enables automatic blood flow velocity computation *in vivo*.**

***Index Terms*—Active contour model, B-splines, gradient vector flow, intravital microscopy, snakes, tracking, vessel boundary detection.**

I. Introduction

AUTOMATED tracking of rolling leukocytes (slow moving white blood cells) *in vivo* is an important task in measuring the inflammatory response [1]–[3]. Before individual cells can be tracked and the corresponding velocities measured, the cells need to be detected. One of the key steps in automatic cell detection is the delineation of the vessel boundary within which to search for cells. Accurate delineation of the vessel boundary decreases the computational cost of searching for cells and also eliminates false positives outside the vessel boundary. Additionally, the vessel boundaries can also be used to measure the diameter of the vessel, which has important application in microvascular research, such as measuring the mean blood flow velocity and the wall shear rate [4], [5]. Due to vasodilation, these venule boundaries are changing on both a spatial and temporal basis. And, because the images obtained *in vivo* have poor contrast and are cluttered and noisy, simple edge detection methods are not effective in the detection of the vessel boundary. This paper tackles vessel boundary detection and tracking using active contour model/snake.

A snake is a parameterized contour [6] that translates and deforms on the image plane according to the strength of the image edges and the internal properties of the contour such as smoothness. Since the original active contour model was introduced by Kass *et al.* [6], the active contour model has been applied successfully in many fields, including medical image segmentation and tracking [2], [7], [8].

Beginning with the initial parametric snake model [6], there have been several advances in designing robust snakes for image analysis. One such advance is the use of gradient vector flow (GVF) in propelling a snake toward the boundary of an object [9], [10]. The basic premise of GVF is the use of vector diffusion to create a field of vectors that guide the snake to a particularly strong boundary. GVF snakes have found wide application in different fields. However, the traditional GVF snake [9], [10] based on the local gradients can be sensitive to noise and clutter. In order to improve the robustness of vessel wall detection and tracking, a multiscale GVF (MSGVF) snake is developed in this paper. The multiscale model is especially useful for vessel boundary detection and tracking, as the images obtained by *in vivo* imaging are very noisy in many cases.

Another contribution found in this paper is the combination of the GVF force with the B-spline snake. B-spline snakes have been successfully applied to tracking problems [11]. B-spline snakes have several characteristics which make them well suitable for describing vessel boundary as well as snake evolution: 1) The B-spline implicitly incorporates contour smoothness and avoids the *ad hoc* tension and rigidity parameters of the traditional GVF snake [12], [13]. 2) Fewer sample points are required to implement contour evolution for the B-spline, and thus, the evolution efficiency is improved [13].

In the next section we will introduce our algorithm for vessel boundary detection and tracking. In this section, we first use frame differencing to estimate the boundary coarsely, similar to the method used in [3]. Then, MSGVF snakes are introduced in order to obtain the accurate location of the vessel boundary. Experimental results and discussion are presented in Sections III and IV, respectively. Finally, conclusions are given in Section V.

II. Materials and Methods

A. Data Acquisition System and Data Collection

The image sequences were acquired using an intravital imaging protocol developed at the University of Virginia. In the *in vivo* imaging system, transilluminated observations were recorded on an intravital microscope (Axioskop; Zeiss,

Manuscript received December 9, 2001; revised May 26, 2003. This work was supported in part by the National Institutes of Health (NIH) under Grant HL68510 and in part by the Whitaker Foundation under Grant RG-01–0112. *Asterisk indicates corresponding author.*

J. Tang is with the Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904-4743 USA.

*S. T. Acton is with the Department of Electrical and Computer Engineering and Department of Biomedical Engineering, University of Virginia, 351 McCormick Road Charlottesville, VA 22904 USA (e-mail: acton@virginia.edu).

Digital Object Identifier 10.1109/TBME.2003.820374

Thornwood, NY) with saline immersion objectives (SW 20/0.5 numerical aperture, SW 40/0.75, and SW 63/0.9). A charge-coupled device camera was connected to the microscope and to a video recorder to obtain the analog video sequences of the vessel. The analog video was then digitized to digital video. When digitizing, the frame rate was set to 30 frames/s. After digitizing, the image sequence was saved in *avi* format on the hard disk of a PC.

The intravital imaging system described above is used to collect data for our experiments. In the experiments, C57BL/6 wild-type mice were obtained from Jackson Laboratories, Bar Harbor, ME. All animal experiments were approved by the institutional animal care and use committee. The cremaster muscle was prepared for intravital microscopy as described in [14]. Venules with diameters between 20 and 40 μm were observed. For each sequence, 300 frames ($=10$ s) were used to test the algorithm.

B. Coarse Detection of Vessel Boundary by Frame Difference

The initial estimation of the vessel boundary position is obtained by the successive frame difference within the video sequence, as in [3]. The frame difference used in this paper is the accumulated absolute frame differences over several frames. Let the image size be denoted as N by M, the intensity value at (x, y) at the kth frame as $I_k(x, y)$; then, the accumulated frame difference in successive K frames starting from the nth frame is defined as

$$\mathrm{diff}(n) = \frac{1}{K}\sum_{k=n}^{n+K-1} \mathrm{diff}(k, k+1) \tag{1}$$

where

$$\mathrm{diff}(k, k+1) = \frac{1}{N \times M}\sum_{x=0}^{N-1}\sum_{y=0}^{M-1} |I_{k+1}(x, y) - I_k(x, y)| \tag{2}$$

is the frame difference between the kth frame and the $(k+1)$th frame. K must be selected so that one entire respiratory cycle is represented. For our murine model, K must be set such that $K \geq 20 (= 2/3\ s)$.

Using the difference obtained by (1), we can find a coarse segmentation of vessel. However, there are small imperfections in the coarse segmentation. Generally, an uncertainly of 5–8 μm is typical (which is on the order of a leukocyte diameter). From the coarse region identified via frame differing, morphological operators are used [15] to remove spurious pixels, eliminate the small and thin objects, breaks objects at thin points, and perform smoothing the boundaries of large objects. Then the connected component labeling technique [15] is applied to the image and the largest motion block is retained as the coarse estimation of the vessel interior. The boundary can be extracted by simply finding the level lines (the 0–1 boundaries) of the connected component.

The initial coarse segmentation is used as the initial image in locating the vessel boundary. As mentioned, the coarse boundary has flaws due to stationary objects (adherent leukocytes), jitter at the boundary, and clutter. To refine the boundary and to track the boundary through the video sequence, the active contour model discussed in the following section is applied.

C. MSGVF Snake

In this section, we will first introduce the existing GVF snake and then describe two improvements to the GVF snake.

1) GVF Snake: Motion of the snake is enacted by minimizing an energy functional that is comprised of internal and external forces. Representing the position of a snake parametrically by $\mathbf{r}(s) = (x(s), y(s))$, a generic snake energy functional can be written as [6]

$$E_{snake} = \int_0^1 [E_{\mathrm{int}}(\mathbf{r}(s)) + E_{ext}(\mathbf{r}(s))]ds \tag{3}$$

where E_{int} and E_{ext} represent the internal energy and the external forces, respectively. The internal force keeps the active contour smooth and may be defined as [6]

$$E_{\mathrm{int}} = \frac{(\alpha |\mathbf{r}_s(s)|^2 + \beta |\mathbf{r}_{ss}(s)|^2)}{2}. \tag{4}$$

The external energy is application-based and is used for guiding the active contour toward the image features of interest. For a typical implementation, the external energy term E_{ext} can be defined as $-|\nabla G_\sigma(x, y) * I(x, y)|$, where $I(x, y)$ is the image intensity at (x, y), $G_\sigma(x, y)$ is a 2D Gaussian kernel with standard deviation σ, and ∇ is the gradient operator. We will instead utilize a *gradient vector flow* model for the external force, in order to increase the capture range of the snake.

Variational calculus techniques are employed to minimize the energy functional in (3), leading to the following Euler equations:

$$\alpha x_{ss} - \beta x_{ssss} - \frac{\partial E_{ext}}{\partial x} = 0 \tag{5}$$

$$\alpha y_{ss} - \beta y_{ssss} - \frac{\partial E_{ext}}{\partial y} = 0. \tag{6}$$

Replacing the external forces $(-\partial E_{ext}/\partial x, -\partial E_{ext}/\partial y)$ (in the x and y directions, respectively) with a vector field (u, v), we have [9], [10]:

$$\alpha x_{ss} - \beta x_{ssss} + u(x(s), y(s)) = 0 \tag{7}$$

$$\alpha y_{ss} - \beta y_{ssss} + v(x(s), y(s)) = 0. \tag{8}$$

The (u, v) terms at a given position (x, y) are determined by descending on the following energy functional [9], [10]

$$\begin{aligned} &E_{GVF}(u, v) \\ &= \iint g(|\nabla f|)(u_x^2 + u_y^2 + v_x^2 + v_y^2)dxdy \\ &\quad + \iint (1 - g(|\nabla f|))((u - f_x)^2 + (v - f_y)^2)dxdy. \end{aligned} \tag{9}$$

In (9)

$$f(x, y) = |\nabla G_\sigma(x, y) * I(x, y)| \tag{10}$$

and

$$g|\nabla f| = \exp\left(-\frac{|\nabla f|}{L}\right). \tag{11}$$

Here, L is a positive constant which is used to control the smoothness of the resulting vector field. The vector field "guides" the snake to the major boundaries.

The solution to (7)–(8) can be obtained explicitly by the following iteration equation:

$$x_t(s,t) = \alpha x_{xx} - \beta x_{ssss} + u(x(s,t), y(s,t)) \quad (12)$$

$$y_t(s,t) = \alpha y_{xx} - \beta y_{ssss} + v(x(s,t), y(s,t)) \quad (13)$$

where in these equations the active contour $\mathbf{r}(s) = (x(s), y(s))$ is treated as a function of time t as well as s : $\mathbf{r}(s,t) = (x(s,t), y(s,t))$.

In digital image processing, the discrete versions of (12) and (13) are used. Let the contour be discretized into n points/snaxels as (x_i, y_i), for $i = 0, 1, \ldots n-1$, and the collection of the coordinates of the snaxels be denoted by (X, Y). Here, $X = [x_0, \ldots, x_{n-1}]^T$ and $Y = [y_0, \ldots, y_{n-1}]^T$. Corresponding to (X, Y), GVF is used to construct the other two vectors

$$U(X,Y) = [u(x_0, y_0), \ldots, u(x_{n-1}, y_{n-1})]^T$$
$$\text{and } V(X,Y) = [v(x_0, y_0), \ldots, v(x_{n-1}, y_{n-1})]^T.$$

Thus, the discrete versions of (12) and (13) can be expressed as follows [16]:

$$X^{t+1} = X^t - AX^t + U(X^t, Y^t) \quad (14)$$

$$Y^{t+1} = Y^t - AY^t + V(X^t, Y^t) \quad (15)$$

where

$$A = \begin{bmatrix} c & b & a & & & a & b \\ b & c & b & a & & & a \\ a & b & c & b & a & & \\ & & & \ddots & & & \\ & & a & b & c & b & a \\ a & & & a & b & c & b \\ b & a & & & a & b & c \end{bmatrix} \quad (16)$$

is an $n \times n$ symmetric pentadiagonal matrix with $a = \beta$, $b = -(4\beta + \alpha)$ and $c = 6\beta + 2\alpha$. $(X^t Y^t)$ and $(X^{t+1} Y^{t+1})$ are the values of (X, Y) at times t and $t+1$, respectively. $(U(X^t, Y^t)$, $V(X^t, Y^t))$ and $(U(X^{t+1}, Y^{t+1})$, $V(X^{t+1}, Y^{t+1}))$ are the values of $U(X, Y)$ and $V(X, Y)$ at times t and $t+1$, respectively. To avoid problems small time steps in implementation, an implicit discretization scheme can be used equivalently [2].

2) Combining GVF With the B-Spline Snake: The traditional active contour model requires the selection of several parameters. In order to address this shortcoming, a B-spline is used here to represent the active contour for vessel boundary detection. Because continuity is implicit in the B-spline model, we do not need to set *ad hoc* weights for tension and rigidity (the parametric snake internal smoothness parameters). Besides, using B-spline to represent the contour, we can employ fewer points on the contour to implement evolution. In this paper, the cubic B-spline model is used. Suppose that the active contour $\mathbf{r}(s) = (x(s), y(s))(0 \le s \le 1)$ is a closed curve and $(C_0 \ldots C_N)$ (where $C_N = C_0$) are a set of control points, where $C_i = (cx_i, cy_i)$ are points sampled from the active contour by uniform sampling, then the active contour can be approximated by N cubic curve segments. The ith curve segment between C_i and C_{i+1} is defined by the points $C_{i-1}, C_i, C_{i+1}, C_{i+2}$ as follows [17]:

$$\mathbf{r}_i(s) = \frac{1}{6}[s^3 \quad s^2 \quad s \quad 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} C_{i-1} \\ C_i \\ C_{i+1} \\ C_{i+2} \end{bmatrix} \quad (17)$$

where $C_{-1} = C_{N-1}$, $C_{N+1} = C_1$ and $i = 0, 1, \ldots N-1$. Equation (17) can be written in a compact format [17]:

$$\mathbf{r}(s) = \sum_{i=0}^{N-1} B_i(s) C_i \quad (18)$$

where $B_i(s)$ are the B-spline basis functions.

One of the properties of the B-spline is local control [17]. Modification of the position of one control point causes only a small part of the curve to change. This leads to an effective algorithm for computing the motion of the snake under the influence of image forces. For vessel delineation and tracking, this local control is particularly effective, as is demonstrated in the results shown in Section III.

Let $(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})$, where $(x_j, y_j) = (x(s_j), y(s_j)) = \mathbf{r}(s_j)$, be n sample points distributed uniformly around the spline. The corresponding GVF force computed on these sample points are $(u(x_0, y_0), v(x_0, y_0)), \ldots, (u(x_{n-1}, y_{n-1}), v(x_{n-1}, y_{n-1}))$. In the B-spline snake, we assume that the displacement of each sample point caused by the snake evolution is proportional to the force applied at that sample point [17]. Thus, if we denote the new points corresponding to sample points after evoluation by $(\bar{x}_0, \bar{y}_0), \ldots, (\bar{x}_{n-1}, \bar{y}_{n-1})((\bar{x}_j, \bar{y}_j) = (\bar{x}(s_j), \bar{y}(s_j)) = \bar{\mathbf{r}}(s_j))$, they can be obtained by

$$\bar{x}_j = x_j + \eta u(x_j, y_j) \quad (19)$$

and

$$\bar{y}_j = y_j + \eta v(x_j, y_j) \quad (20)$$

where $j = 0, \ldots, n-1$. The new sample points $(\bar{x}_0, \bar{y}_0), \ldots, (\bar{x}_{n-1}, \bar{y}_{n-1})$ are used to compute new control points using least squares estimation by minimizing [17]

$$\hat{\vec{\mathrm{C}}} = \arg\min_{\vec{\mathrm{C}}} \sum_{i=0}^{n-1} \left\| \bar{\mathbf{r}}(s_i) - \sum_{k=0}^{N-1} B_k(s_i) C_k \right\| \quad (21)$$

where

$$\vec{\mathrm{C}} = (C_0, C_1, \ldots, C_{N-1}).$$

The new control points are used to obtain the new sample points using (17). These computations are repeated until convergence.

3) Multiscale Gradient Vector Flow (MSGVF) Snake: In order to make the snake more robust to the noise, small scale detail and clutter, this paper introduces a MSGVF. The MSGVF is based on scale space theory.

Scale space theory has found wide application in image processing. The basic idea of scale space theory is concerned with the generation, from a single initial image $I_0(x, y)$, of as sequence of simplified images $I_\lambda(x, y)$, where $I_\lambda(x, y)$ appears to be rougher, coarser version of $I_0(x, y)$ as λ increases [18]. From the scale space representation, we can compute the edge

map and obtain the multiscale edge map of the original image by

$$A_\lambda(x,y) = |\nabla I_\lambda(x,y)|. \tag{22}$$

The multiscale edge map $\{A_\lambda\}$ can be used to compute the MSGVF. Replacing f by A_λ in (9), we obtain the GVF of the image $I_0(x,y)$ in scale λ by minimizing the following equation:

$$\begin{aligned} &E_{GVF}(u,v,\lambda) \\ &= \iint g(|\nabla A_\lambda|)(u_x^2+u_y^2+v_x^2+v_y^2)dxdy \\ &\quad + \iint (1-g(|\nabla A_\lambda|))((u-(A_\lambda)_x)^2+(v-(A_\lambda)_y)^2)dxdy. \end{aligned} \tag{23}$$

The GVF of the image $I_0(x,y)$ at scale λ is then denoted by $(u_\lambda(x(s),y(s)), v_\lambda(x(s),y(s))$. This set of forces is called the MSGVF of the image $I_0(x,y)$.

To obtain MSGVF, we first compute the multiscale representation of the original image. There are many ways to produce a multiscale representation of the original image. In this approach, we use a series of low-pass filters. Let $\{h_\lambda(x,y)\}$ be the unit pulse response of a low-pass filter corresponding to scale λ. Then $I_\lambda(x,y)$ can be obtained by

$$I_\lambda(x,y) = \sum_m \sum_n h_\lambda(m,n) I_0(x-m, y-n). \tag{24}$$

Although the Gaussian filter [19] can also be used as the scale generating mechanism, in which

$$h_\lambda(x,y) = \frac{1}{2\pi\lambda^2} e^{-(x^2+y^2)/2\lambda^2} \tag{25}$$

we employ average filters [19] for computational simplicity. The unit pulse response for the average filter at scale λ is given by

$$h_\lambda(x,y) = \begin{cases} \frac{1}{(2\lambda+1)^2} & -\lambda \le x,y \le \lambda \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

where $(2\lambda+1)\times(2\lambda+1)$ is the size of the kernel. Note that the average filter requires $(2\lambda+1)^2-1$ adds and one multiply for a single result at location (x,y), while the same computation for a (truncated) Gaussian kernel is $(2\lambda+1)^2-1$ adds and $(2\lambda+1)^2$ multiplies.

Now let us introduce our MSGVF B-spline snake. Suppose that an M-scale GVF is adopted in the evolution, then the algorithm is described as follows:

1. Given a sequence of scales $\lambda_0, \lambda_1, \ldots, \lambda_{M-1}$, where $\lambda_{i+1} = \lambda_i + \omega$ and ω represents the granularity in detail in units of pixel width. In our implementation of vessel boundary detection, $\omega = 3$. (Using a finer granularity, $\omega = 2$, did not result in a lower RMSE.)
2. Let $k = M-1$ and initialize the active contour.
3. Sample the active contour every P pixels (typical $P = 8$), the obtained sample points denoted by $(x_0^k, y_0^k), \ldots, (x_{n-1}^k, y_{n-1}^k)$.
4. At scale λ_k, use **(24)** to generate $I_\lambda(x,y)$ and use **(22)** to get A_{λ_k}.
5. Perform gradient descent on **(23)** to obtain the GVF at scale λ_k.
6. Evolve contour using **(19)** and **(20)**, starting from initial samples $(x_0^k, y_0^k), \ldots, (x_{n-1}^k, y_{n-1}^k)$. After the evolution converges, the obtained points are denoted by $(\bar{x}_0^k, \bar{y}_0^k), \ldots, (\bar{x}_{n-1}^k, \bar{y}_{n-1}^k)$.
7. Using least squares method by minimizing **(21)** to obtain control points from points $(\bar{x}_0^k, \bar{y}_0^k), \ldots, (\bar{x}_{n-1}^k, \bar{y}_{n-1}^k)$. The control points are used to build a new active contour using **(17)**.
8. Reparameterization.
9. If $k=0$ stop. Else set $k=k-1$ and proceed to Step 3.

Fig. 1 shows the GVFs of a noisy image in different scales and a comparison between the multiscale approach and the standard approach in delineating an object boundary. From synthetic experiments such as the one in Fig. 1, we find that the MSGVF B-spline snake shows promise for detecting contours in the presence of noise and clutter. Fig. 2 shows the GVFs of a vessel image block in different scales.

D. Detection and Tracking of Vessel Boundary: Combining Frame Differencing With the Snake

The detection and tracking of the vessel boundaries are summarized as follows. First, we use the accumulated frame difference to obtain the approximate curve of vessel boundary for the first frame of the video sequence and use the approximate curve as the initial snake to implement snake evolution in the first frame. The evolution of the snake is implemented using the MSGVF B-spline snake developed in Section II-C. After the snake has converged, the final contour obtained in the first frame is used as the initial snake in the second frame and the MSGVF B-spline snake is used to perform evolution. This processing is repeated until boundaries of the vessels within all of the frames are obtained.

III. Experimental Results

A. Ground Truth and Evaluation Metric

After obtaining the video sequences, we collect ground truth data for comparison. The average of manually marked boundaries (by several technicians) serves as ground truth in this case. A sampling of the frames (every 80 frames in this case) in each sequence is manually processed. For each selected frame (and from each observer), we obtain two ground truth curves, one corresponding to the upper vessel boundary and the other corresponding to the lower vessel boundary. The metric adopted in this paper for comparison is the root mean squared error (RMSE) in units of micrometers. Let the coordinates of the points from the curve which fits the vessel boundary obtained by our algorithm are $(x_0,y_0), \ldots, (x_{n-1}, y_{n-1})$. For the point (x_k, y_k) $(k = 0, \ldots n-1)$, we found the corresponding point $(\bar{x}_k, \bar{y}_k)$ $(k = 0, \ldots n-1)$ on the ground truth curves that has

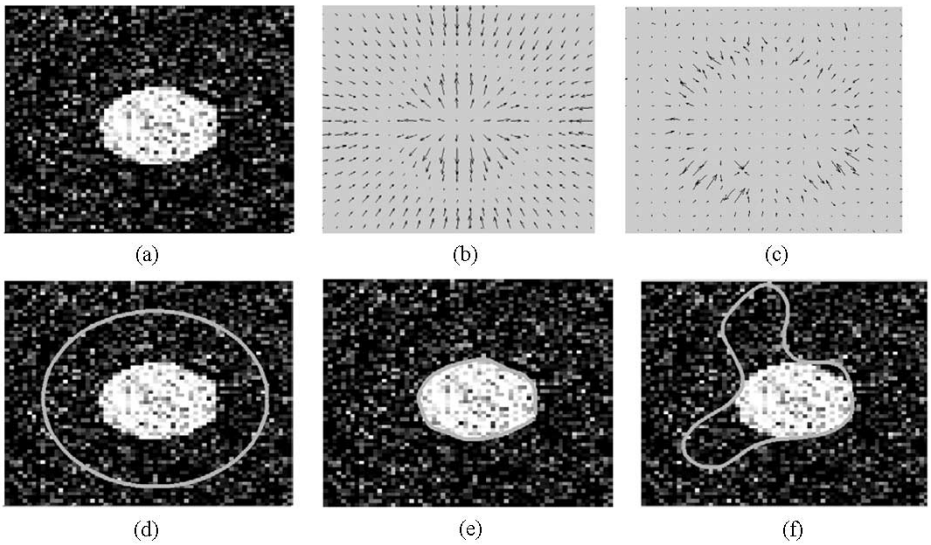

Fig. 1. GVF for the original noisy image in different scales and tracking results using MSGVF and GVF, respectively. Notes: For visual need, we enlarged the GVF vectors of the images and did some sampling. (a) Original noisy image. (b) Scale $\lambda = 6$. (c) Scale $\lambda = 0$. (d) Initialization of contour. (e) Delineated by MSGVF. (f) Delineated by traditional GVF.

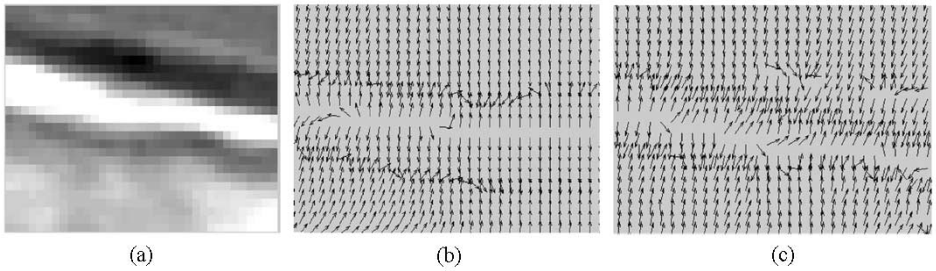

Fig. 2. GVF of an original vessel image block in different scales. (a) Enlarged original vessel block. (b) Scale $\lambda = 6$. (c) Scale $\lambda = 0$.

the closest distance from point (x_k, y_k) $(k = 0, \ldots n-1)$. Then the RMSE is computed as follows:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{k=0}^{n-1} d_k^2}{n}} \tag{27}$$

where

$$d_k = \sqrt{(x_k - \bar{x}_k)^2 + (y_k - \bar{y}_k)^2}. \tag{28}$$

B. Detection and Tracking Results

For each sequence, we first used the method in Section II-B to obtain the accumulated difference and then used simple edge detection method to find the coarse estimation of the boundary. The obtained contour was used as the initial active contour for detecting vessel boundary within the first frame using the snake developed in Section II-C. After we computed the vessel boundary for the first frame, we used it as the initial active contour for the second frame and used it to detect the vessel boundary in the second frame. This processing continues until the boundaries in all of the frames have been detected.

The detection and tracking results of two sequences are shown in Figs. 3 and 4, respectively. Fig. 3 shows the results for a "TNF-α sequence" and Fig. 4 shows the results for a "Trauma sequence." With the trauma sequence, an inflammatory response is induced by the surgical incision used to expose the venules under observation. For the TNF-α sequence, a cytokine, TNF-α (tumor necrosis factor alpha), is used to stimulate a more pronounced inflammatory response. With the TNF-α sequence, we observe a greater number of rolling leukocytes (higher rolling leukocyte flux), increased vasodilation (affecting the venule diameter), and slower rolling velocities for the cells. Thus, the TNF-α sequence should provide a challenging test for the vessel detection and tracking algorithm given increased clutter and increased vessel dilation. Fig. 3(a) shows the initial active contour obtained for the first frame and Fig. 3(b), (c), (d), and (e) shows the detection results in the 1st, 81st, 161st, and 241st frames, respectively. Fig. 4(a)–(d) shows the detection results in the 1st, 81st, 161st, and 241st frames, respectively, of the trauma sequence. The number of the control points was set to be 8. The sample points were obtained every 8 pixels. These numbers were determined empirically.

The size of the largest window to perform average filters were chosen to be less than one quarter of the diameter of the vessel, which can be estimated coarsely from the coarse segmentation of the vessel. In the experiments using the TNF-α sequence, the minimum vessel diameter is approximately 78 in units of pixel width, so the scale space utilized scales $\lambda = 0$, 3, 6, 9. In the experiments with the trauma sequence, the minimum vessel diameter is about 47 in units of pixel width, so the scale space progressed with scales $\lambda = 0$, 3, 6.

The RMSE values computed between the ground truth data and the results obtained by the automated methods are shown in Table I for the "TNF-α sequence" and in Table II for "Trauma

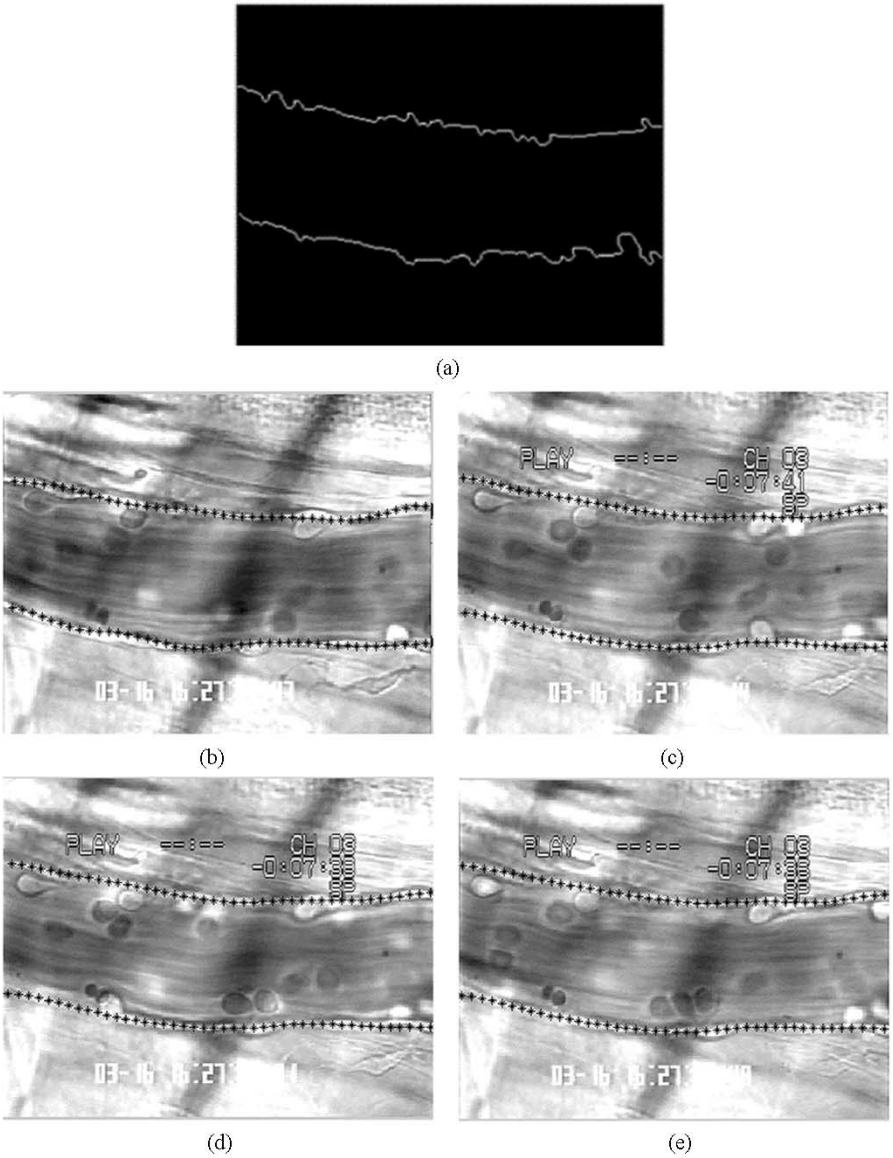

Fig. 3. Boundary detection results obtained with the proposed algorithm on TNF-α sequence. (a) Coarse detection of Vessel Boundary by frame differencing. (b) Final active contour at the first frame. (c) Final active contour at the 81st frame. (d) Final active contour at the 181st frame. (e) Final active contour at the 241st frame.

sequence." Here, we compare the GVF-driven spline snake with the MSGVF-driven spline snake. From the tables, we can see that the multiscale method yields lower RMSE values compared to those of the standard GVF spline snake for both sequences. In the experiments, the errors were obtained by comparison with the curve computed by averaging the manual results obtained by seven technicians.

In addition to computing the RMSE, we also asked the technicians to visually check the detection and tracking results. 10 sequences were used in the visual validation experiments. For all of the 10 sequences, tracking was successful over all the frames and there are no large ($> 5\ \mu\mathrm{m}$) tracking errors given by the MSGVF-driven B-spline snake. For GVF-driven B-spline snake, the tracking was successful for only one sequence. For all the other sequences, the vessel boundary was "lost" during each sequence.

Besides computing the tracking accuracy, we also computed the time needed delineate the vessel. The computational cost of the active contour tracking is proportional to the jitter /movement of the sequence. Due to the increased jitter/movement in the trauma sequence, the active contour also experiences increased translational movement to continuously capture the vessel boundary. Therefore, the trauma sequence has higher expense performance. The average time needed to delineate the vessel sequence for the TNF-α sequence is 1.5 s per frame and for trauma sequence, the expense is 3.1 s per frame. These timing measurements were taken using a 2 GHz Pentium IV PC with 1 GB of RAM.

C. Application: Automatic Computation of Wall Shear Rate

Using the tracking results of the vessel boundary, we can obtain an automatic computation method of mean blood flow velocity and wall shear rate. We first use the vessel boundary to find the centerline. For each point on the one boundary of the vessel, we find a corresponding point on the other boundary that

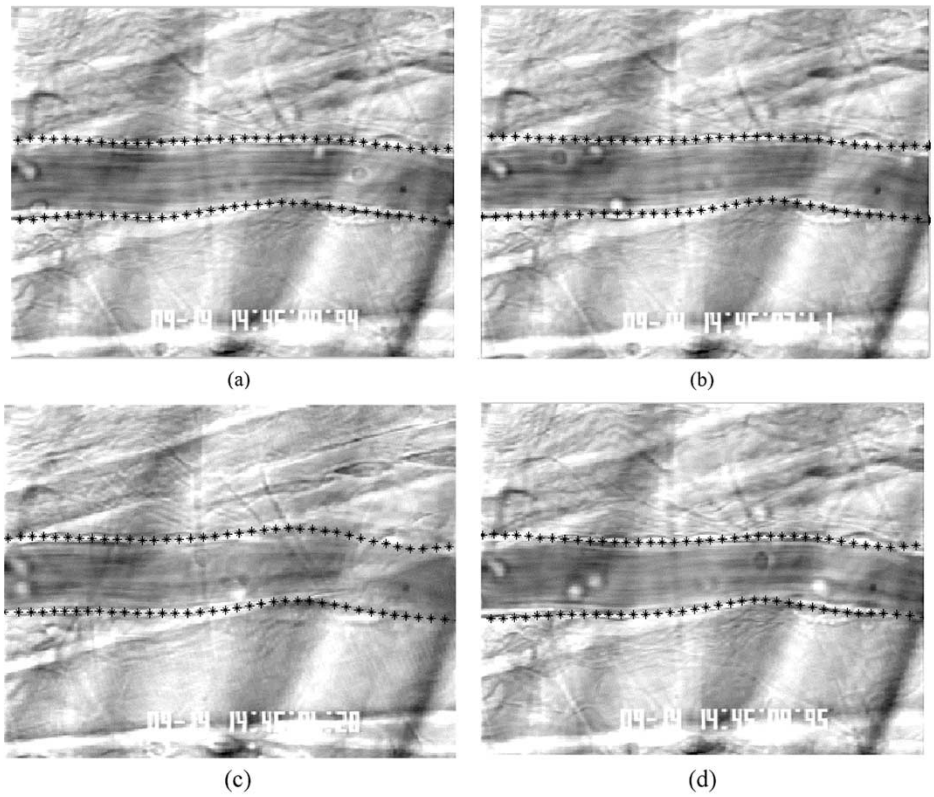

Fig. 4. Boundary detection results obtained with the proposed algorithm on the trauma sequence. (a) Final active contour at the first frame. (b) Final active contour at the 81st frame. (c) Final active contour at the 181st frame. (d) Final active contour at the 241st frame.

TABLE I
RMSE FOR TNF-α SEQUENCE

Method	RMSE (microns)			
	Frame 1	Frame 81	Frame 161	Frame 241
GVF Spline Snake	2.6	3.3	3.3	3.3
Multiscale GVF Spline Snake	1.9	2.3	2.0	1.9

TABLE II
RMSE FOR TRAUMA SEQUENCE

Method	RMSE(microns)			
	Frame 1	Frame 81	Frame 161	Frame 241
GVF Spline Snake	3.5	3.6	4.1	4.3
Multiscale GVF Spline Snake	1.5	1.5	1.9	1.9

TABLE III
MEAN BLOOD FLOW VELOCITIES COMPUTED IN DIFFERENT EXPERIMENTS FOR A VIDEO SEQUENCE OBTAINED *IN VIVO*

Experiment	1	2	3	4
Blood Flow Velocity	1.02mm/s	0.988mm/s	0.985mm/s	0.989mm/s

has the closest distance from this point. For each pair of points, we compute the midpoint of the line segment connecting them. The curve constructed by these midpoints is the centerline. The diameter can be estimated by averaging the distances between the corresponding pairs of boundary points.

The spatial correlation method developed in [4] is used to compute the mean blood flow velocity. The estimated values in different times for an *in vivo* video sequence are shown in Table III. The actual blood flow velocity measured by the system described in [20] is 1 mm/s.

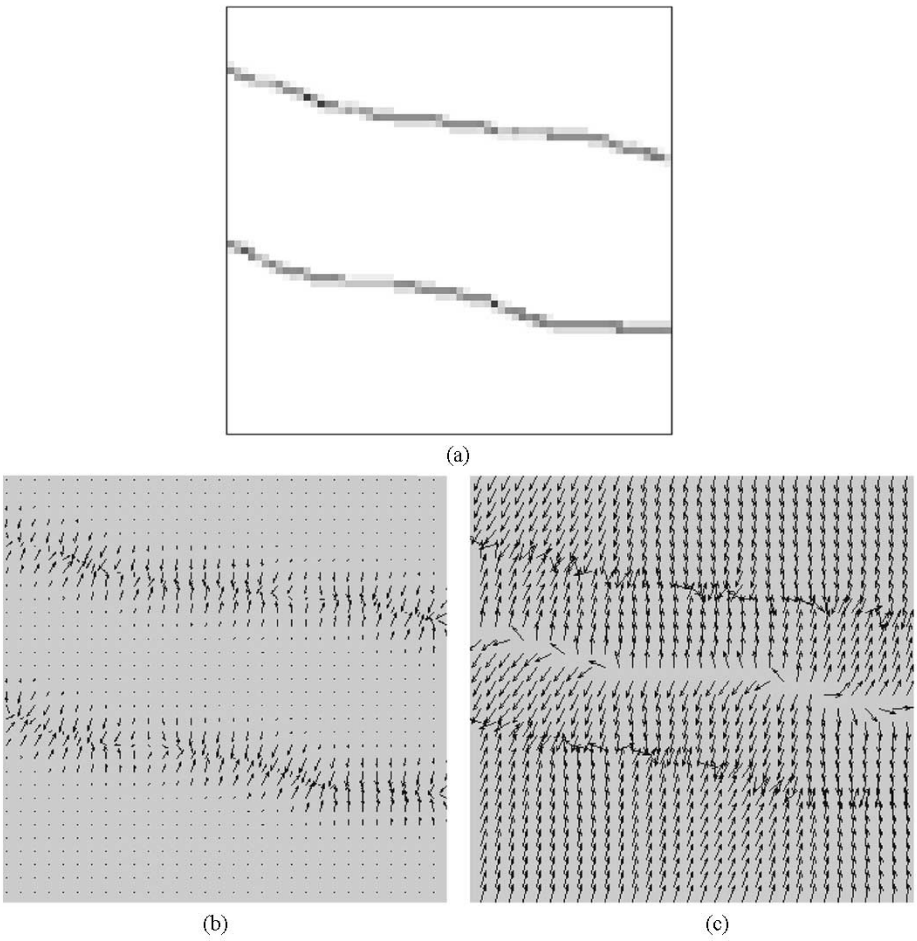

Fig. 5. Streamline originating from a 64 × 64 array similar to vessel boundary. (a) Original 64 × 64 array. (b) Traditional potential force field. (c) GVF force field.

Let the mean blood flow velocity be V_B, then the wall shear rate can be estimated as

$$\gamma_W = 2.12 \times 8 \times \frac{V_B}{d} \tag{32}$$

where d is the vessel diameter and 2.12 is an empirical correction factor based on velocity profiles measured *in vivo* [5]. The computation of wall shear rate used the same sequence for the measuring of the blood flow velocity. The wall shear rate was 746 mm/s. Here, the blood flow velocity V_B was obtained by averaging the four velocities in Table III. Without the automated delineation of the vessel wall boundary, the wall shear rate could not be computed automatically.

IV. Discussion

GVF has been used as an external force to drive the active contour to the vessel boundary in this paper. The effectiveness of the GVF force can be observed from the streamlines (using the motivating arguments of [9]) shown in Fig. 5. From Fig. 5, we can find that, compared with the traditional potential forces, the GVF capture range exceeds that of the traditional gradient-based external force model. This property is especially useful for vessel boundary tracking as the initialization of the snake (from frame averaging) may be distant from the real boundary due to jitter.

However, because of the noise, the snake based on a single-scale GVF is not robust. In order to enhance the robustness of traditional GVF, MSGVF is introduced to drive the snake. In MSGVF, the first step is the computation of the multiscale edge map. There are many ways to obtain the multiscale edge map and we employed a succession of average filters of increasing scale. These filters approximate a Gaussian scale space [21]. Although Gaussian filters can be used to obtain the multiscale edge map [18], the results from using the more expensive Gaussian filter instead of the average filter were not significantly improved (the difference in RMSE was less than 0.1 μm).

With the GVF snake (14)–(15), the choice of the tension and rigidity parameters is very important. However, there are not general methods to perform this parameter estimation. The combining of B-snake and GVF snake can solve this problem. In the new algorithm, the smoothness is implicitly built into the model and avoids the *ad hoc* tension and rigidity parameters of the traditional GVF snakes. The second advantage is that the proposed snakes combing B-spline and GVF uses much less points to perform evolution than traditional GVF snakes and, thus, it improves the evolution velocity.

V. Conclusion

This paper studies vessel boundary detection and tracking from intravital imaging of the microvasculature. Because intravital imaging is not ideal and the images obtained are typically cluttered, the traditional techniques such as simple edge detection are not effective in the detection of the vessel boundary. This paper proposes a novel vessel boundary detection and tracking algorithm. Our algorithm uses a MSGVF snake. We have compared our algorithm with the traditional GVF snake on both synthetic images and real microscopic images obtained *in vivo*. Experiments show that the MSGVF snakes reduces the RMSE yielded by the fixed-scale GVF snake. In turn, the success of automated vessel boundary detection enables automatic computation of wall shear rate.

References

[1] S. T. Acton, K. Wethmar, and K. Ley, "Automatic tracking of rolling leukocytes *in vivo*," *Microvasc. Res.*, vol. 63, pp. 139–148, 2002.

[2] N. Ray, S. T. Acton, and K. Ley, "Tracking leukocytes *in vivo* with shape and size constrained active contours," *IEEE Trans. Med. Imag. (Special Issue on Image Analysis in Drug Discovery and Clinical Trials)*, vol. 21, pp. 1222–1235, Oct. 2002.

[3] Y. Sato, J. Chen, R. A. Zoroofi, N. Harada, S. Tamura, and T. Shiga, "Automatic extraction and measurement of leukocyte motion in microvessels using spatiotemporal image analysis," *IEEE Trans. Biomed. Eng.*, vol. 44, pp. 225–236, Apr. 1997.

[4] A. R. Pries, "A versatile video image analysis system for microcirculatory research," *Int. J. Microcirc.: Clin. Exp.*, vol. 7, pp. 327–345, 1988.

[5] R. S. Reneman, B. Woldhuis, M. G. A. oude Egbrind, D. W. Slaff, and G. J. Tangelder, "Concentration and velocity profiles of blood cells in the microcirculation," in *In Advances in Cardiovascular Engineering*, N. Hwang, V. Turitto, and M. Yen, Eds. New York: Plenum, 1992, pp. 25–40.

[6] M. Kass, A. Witkin, and D. Terzopolous, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, pp. 321–331, 1987.

[7] P. Bamford and B. Lovell, "Unsupervised cell nucleus segmentation with active contours," *Signal Processing*, vol. 71, pp. 203–213, 1998.

[8] D. Chung and G. Sapiro, "Segmenting skin lesions with partial-differential-equation-based image processing algorithms," *IEEE Trans. Med. Imag.*, vol. 19, pp. 763–767, July 2002.

[9] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Trans. Image Processing*, vol. 7, pp. 359–369, Mar. 1998.

[10] ——, "Generalized gradient vector flow external force for active contours," *Signal Processing*, vol. 71, pp. 131–139, 1998.

[11] A. Blake and M. Isard, *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Berlin, Germany: Springer-Verlag, 1998.

[12] P. Brigger, J. Hoeg, and M. Unser, "B-spline snakes: A flexible tool for parametric contour detection," *IEEE Trans. Image Processing*, vol. 9, pp. 1484–1496, Sept. 2000.

[13] A. K. Klein, F. Lee, and A. A. Amini, "Quantitative coronary angiography with deformable spline models," *IEEE Trans. Med. Imag.*, vol. 16, pp. 468–482, Oct. 1997.

[14] K. Ley, D. C. Bullard, M. L. Arbones, R. Bosse, D. Vestweber, T. F. Tedder, and A. L. Beaudet, "Sequential contribution of L- and P-selectin to leukocyte rolling *in vivo*," *J. Exp. Med.*, vol. 181, pp. 669–675, 1995.

[15] S. Marchand-Maillet and Y. M. Sharaiha, *Binary Digital Image Processing: A Discrete Approach*. New York: Academic, 2000.

[16] N. Ray, B. Chanda, and J. Das, "A fast and flexible multiresolution snake with a definite termination criterion," *Pattern Recogn.*, vol. 34, pp. 1483–1490, 2001.

[17] B. Levienaise-Obadia and A. Gee, "Adaptive segmentation of ultrasound images," presented at the *8 th British Machine Vision Conf.*, Essex, U.K., 1997.

[18] J. M. Morel and S. Solimini, *Variational Methods in Image Segmentation*. Basel, Switzerland: Birkhauser, 1994.

[19] A. C. Bovik and S. T. Acton, "Basic linear filtering with application to image enhancement," in *Handbook of Image and Video Processing*. New York: Academic, 2000, pp. 71–79.

[20] R. N. Pittman and M. L. Ellsworth, "Estimation of red cell flow in microvessels: Consequences of the Baker-Wayland spatial averaging model," *Microvasc. Res.*, vol. 32, pp. 371–388, 1986.

[21] W. M. Wells, "Efficient synthesis of Gaussian filters by cascaded uniform filters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 234–239, 1986.

Jinshan Tang (M'01–SM'03) received the Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing P.R. China, in 1998.

From September 1998 to June 2000, he was an Invited Researcher with ATR Media Integration and Communications Research Laboratories (MIC), Kyoto, Japan. In June 2000, he went to Harvard University, Cambridge, MA, to work on image enhancement technology. Since December 2001, he has been a Research Scientist in Department of Electrical and Computer Engineering, University of Virginia, Charlottesville. His research interests are smart computers, content-based image retrieval, cell detection and tracking, face detection and recognition, and image enhancement for low-vision.

Scott T. Acton (S'89–M'93–SM'99) received the B.S. degree in electrical engineering from Virginia Polytechnic Institute and State University Blacksburg, in 1988. He received the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas at Austin in 1990 and 1993, respectively.

He has worked in industry for AT&T, the MITRE Corporation, and Motorola, Inc. and in academia for Oklahoma State University, Stillwater.

For his research in video tracking, Dr. Acton received an ARO Young Investigator Award. He received the Halliburton Outstanding Young Faculty Award in 1998. In 1997, he was named the Eta Kappa Nu Outstanding Young Electrical Engineer – a national award that has been given annually since 1936. At the University of Virginia, he was named the Outstanding New Teacher in 2002, elected a faculty fellow in 2003, and is the Walter N. Munster chair. He is an active participant in the SPIE and the IEEE, served as Associate Editor for the IEEE Transactions on Image Processing and currently serves as Associate Editor the IEEE Signal Processing Letters.