



ELSEVIER

Signal Processing 71 (1998) 131–139

**SIGNAL
PROCESSING**

Generalized gradient vector flow external forces for active contours¹

Chenyang Xu, Jerry L. Prince^{2,*}

Image Analysis and Communications Laboratory, Department of Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD 21218, USA

Abstract

Active contours, or snakes, are used extensively in computer vision and image processing applications, particularly to locate object boundaries. A new type of external force for active contours, called *gradient vector flow* (GVF) was introduced recently to address problems associated with initialization and poor convergence to boundary concavities. GVF is computed as a diffusion of the gradient vectors of a gray-level or binary edge map derived from the image. In this paper, we generalize the GVF formulation to include two spatially varying weighting functions. This improves active contour convergence to long, thin boundary indentations, while maintaining other desirable properties of GVF, such as an extended capture range. The original GVF is a special case of this new generalized GVF (GGVF) model. An error analysis for active contour results on simulated test images is also presented. © 1998 Elsevier Science B.V. All rights reserved.

Zusammenfassung

Aktive Umriss- oder Schlangen, werden vielfach in Computervision- und Bildverarbeitungs-Anwendungen benutzt, um insbesondere Objektgrenzen zu lokalisieren. Ein neuer Typ äußerer Kräfte für aktive Umriss-*Gradient Vector Flow* (GVF) genannt, wurde kürzlich eingeführt, um Probleme anzusprechen, die mit Initialisierung und schlechter Konvergenz zu Grenzkonkavitäten zusammenhängen. GVF wird als eine Diffusion des Gradientenvektors einer Graustufen- oder 'Binary Edge'-Karte berechnet, die aus dem Bild gewonnen werden. In diesem Artikel verallgemeinern wir die GVF Formulierung, so daß zwei räumlich variierende Gewichtsfunktionen eingeschlossen werden. Dies verbessert die Konvergenz aktiver Umriss- zu langen, dünnen Grenzmarkierungen, während andere wünschenswerte Eigenschaften des GVF, wie erweiterter Empfangsbereich, erhalten bleiben. Das ursprüngliche GVF ist ein Spezialfall dieses neuen verallgemeinerten GVF (GGVF) Modells. Eine Fehleranalyse von Ergebnissen aktiver Umriss- mit simulierten Testbildern wird ebenfalls präsentiert. © 1998 Elsevier Science B.V. All rights reserved.

Résumé

Les contours actifs, ou serpents (snakes), sont utilisés intensivement en vision par ordinateur et pour les applications de traitement d'images, particulièrement pour localiser les contours d'objets. Un nouveau type de force externe pour les

* Corresponding author. Tel.: +1 410 516 5192; fax +1 410 516 5566; e-mail: prince@jhu.edu.

¹ A preliminary version of this paper appeared in the Proceedings of the Johns Hopkins University 1997 Conference of Information Sciences and Systems.

² This research was supported by NSF grant MIP9350336.

contours actifs, appelé *flux de vecteurs gradients* (FVG) a été introduit récemment pour traiter les problèmes associés à l'initialisation et la faible convergence vers des concavités dans les contours. Le FVG est calculé comme une diffusion des vecteurs gradients d'une carte des contours d'une image en niveaux de gris ou binaire. Dans cet article, nous généralisons la formulation du FVG pour y inclure deux fonctions de poids à variation spatiale. Ceci améliore la convergence des contours actifs vers les indentations de contours fines et longues, tout en maintenant les autres propriétés intéressantes des FVG comme la plage de capture étendue. Les FVG originaux sont un cas particulier des modèles de FVG généralisés. Une analyse de l'erreur des résultats de contours actifs sur des images de test synthétiques est aussi présentée. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Edge detection; Image segmentation; Shape representation and recovery; Deformable models; Active contour models; Gradient vector flow

1. Introduction

Active contours, or *snakes*, are curves defined within an image domain, that can move under the influence of internal forces within the curve itself and external forces derived from the image data [9]. The internal and external forces are defined so that the snake will conform to an object boundary or other desired features within an image. Snakes are widely used in many applications, including edge detection [9], shape modeling [7,12,13], segmentation [2,6,10] and motion tracking [10,14].

There are two key difficulties in the design and implementation of active contour models. First, the initial contour must, in general, be close to the true boundary or else it will likely converge to the wrong result. Second, active contours have difficulty progressing into boundary concavities [1,5]. In [15,16], Xu and Prince developed a new external force, called *gradient vector flow* (GVF), which largely solves both problems. GVF is computed as a diffusion of the gradient vectors of a gray-level or binary edge map derived from the image. The resultant field has a large capture range, which means that the active contour can be initialized far away from the desired boundary. The GVF field also tends to force active contours into boundary concavities, where traditional snakes have poor convergence. It still has difficulties, however, forcing a snake into long, thin boundary indentations.

In this paper, we generalize the GVF formulation to include two spatially varying weighting functions. These weighting functions define a tradeoff between

smoothness of the resulting GVF field and its conformity to the gradient of the underlying edge map. The external force fields derived from this new *generalized GVF* (GGVF) improve active contour convergence into long, thin boundary indentations, while maintaining other desirable properties of GVF, such as the extended capture range. The original GVF is a special case of GGVF. An error analysis of active contour results on simulated test images is also presented.

2. Background

2.1. Traditional snakes

A traditional snake is a curve $\mathbf{x}(s) = [x(s), y(s)]$, $s \in [0, 1]$, that moves through the spatial domain of an image to minimize the energy functional

$$E = \int_0^1 \frac{1}{2} (\alpha |\mathbf{x}'(s)|^2 + \beta |\mathbf{x}''(s)|^2 + E_{\text{ext}}(\mathbf{x}(s))) ds, \quad (1)$$

where α and β are weighting parameters that control the snake's tension and rigidity, respectively, and $\mathbf{x}'(s)$ and $\mathbf{x}''(s)$ denote the first and second derivatives of $\mathbf{x}(s)$ with respect to s . The external energy function E_{ext} is derived from the image so that it takes on its smaller values at the features of interest, such as boundaries. Examples of typical external energy functions are $\pm G_\sigma(x, y) * I(x, y)$ for lines and $-\|\nabla(G_\sigma(x, y) * I(x, y))\|^2$ for step edges [3,9], where $I(x, y)$ is a gray-level image, $G_\sigma(x, y)$ is a two-dimensional Gaussian function with standard deviation σ , and ∇ is the gradient operator.

A snake that minimizes E must satisfy the Euler equation

$$\alpha \mathbf{x}''(s) - \beta \mathbf{x}(s) - \nabla E_{\text{ext}} = 0. \quad (2)$$

To find a solution to Eq. (2), the snake is made dynamic by treating \mathbf{x} as function of time t as well as s — i.e., $\mathbf{x}(s, t)$. Then, the partial derivative of \mathbf{x} with respect to t is then set equal to the left-hand side of Eq. (2) as follows:

$$\mathbf{x}_t(s, t) = \alpha \mathbf{x}''(s, t) - \beta \mathbf{x}'''(s, t) - \nabla E_{\text{ext}}. \quad (3)$$

When the solution $\mathbf{x}(s, t)$ stabilizes, the term $\mathbf{x}_t(s, t)$ vanishes and we achieve a solution of Eq. (2).

2.2. GVF snakes

In [15,16], we used Eq. (3) as a starting point to define a new snake, called the *GVF snake*. We proposed to replace the external force term $-\nabla E_{\text{ext}}$ in Eq. (3) with a GVF field $\tilde{\mathbf{v}}(x, y)$ defined as the equilibrium solution of the following system of partial differential equations:

$$\mathbf{v}_t = \mu \nabla^2 \mathbf{v} - (\mathbf{v} - \nabla f) |\nabla f|^2, \quad (4)$$

where \mathbf{v}_t denotes the partial derivative of $\mathbf{v}(x, y, t)$ with respect to t , and $\nabla^2 = \partial^2/\partial x^2 + \partial^2/\partial y^2$ is the Laplacian operator (applied to each spatial component of \mathbf{v} separately). Here f is an edge map derived from the image $I(x, y)$, having the property that it is larger near image edges. This edge map can be either gray-level or binary valued. It can be computed using $\pm G_\sigma(x, y) * I(x, y)$ or $|\nabla(G_\sigma(x, y) * I(x, y))|^2$, or any conventional image edge detector (cf. [8]).

3. Generalized GVF

GVF has many desirable properties as an external force for snakes [15,16]. It still has difficulties, however, forcing a snake into long, thin boundary indentations. We hypothesized that this difficulty could be caused by excessive smoothing of the field near the boundaries, governed by the coefficient μ in Eq. (4). We reasoned that introducing a spatially

varying weighting function, instead of the constant μ , and decreasing the smoothing effect near strong gradients, could solve this problem. In the following formulation, which we have termed *generalized GVF* (GGVF), we replace both μ and $|\nabla f|^2$ in Eq. (4) by more general weighting functions. An alternative generalization, which follows from a variational formulation, is given in Appendix A.

We define GGVF as the equilibrium solution of the following vector partial differential equation:

$$\mathbf{v}_t = g(|\nabla f|) \nabla^2 \mathbf{v} - h(|\nabla f|) (\mathbf{v} - \nabla f). \quad (5)$$

The first term on the right is referred to as the *smoothing term* since this term alone will produce a smoothly varying vector field. The second term is referred to as the *data term* since it encourages the vector field \mathbf{v} to be close to ∇f computed from the data. The weighting functions $g(\cdot)$ and $h(\cdot)$ apply to the smoothing and data terms, respectively. Since these weighting functions are dependent on the gradient of the edge map which is spatially varying, the weights themselves are spatially varying, in general. Since we want the vector field \mathbf{v} to be slowly varying (or smooth) at locations far from the edges, but to conform to ∇f near the edges, $g(\cdot)$ and $h(\cdot)$ should be monotonically non-increasing and non-decreasing functions of $|\nabla f|$, respectively.

The above equation reduces to that of GVF when

$$g(|\nabla f|) = \mu, \quad (6)$$

$$h(|\nabla f|) = |\nabla f|^2. \quad (7)$$

Since $g(\cdot)$ is constant here, smoothing occurs everywhere; however, $h(\cdot)$ grows larger near strong edges, and should dominate at the boundaries. Thus, GVF should provide good edge localization. The effect of smoothing becomes apparent, however, when there are two edges in close proximity, such as when there is a long, thin indentation along the boundary. In this situation, GVF tends to smooth between opposite edges, losing the forces necessary to drive an active contour into this region.

To address this problem, weighting functions can be selected such that $g(\cdot)$ gets smaller as $h(\cdot)$ becomes larger. Then, in the proximity of large gradients, there will be very little smoothing, and the effective vector field will be nearly equal to the

gradient of the edge map. There are many ways to specify such pairs of weighting functions. In this paper, we use the following weighting functions for GGVF:

$$g(|\nabla f|) = e^{-(|\nabla f|/K)}, \quad (8)$$

$$h(|\nabla f|) = 1 - g(|\nabla f|). \quad (9)$$

The GGVF field computed using this pair of weighting functions will conform to the edge map gradient at strong edges, but will vary smoothly away from the boundaries. The specification of K determines to some extent the degree of tradeoff between field smoothness and gradient conformity.

As in GVF [16], the partial differential equation (5) specifying GGVF, can be implemented using an explicit finite difference scheme, which is stable if the time step Δt and the spatial sample intervals Δx and Δy satisfy

$$\Delta t \leq \frac{\Delta x \Delta y}{4g_{\max}},$$

where g_{\max} is the maximum value of $g(\cdot)$ over the range of gradients encountered in the edge map image. While an implicit scheme for the numerical implementations of Eq. (5) would be unconditionally stable and therefore not need this condition, the explicit scheme is faster. Still faster methods – for example, the multigrid method – are possible.

4. Experimental results

In the following experiments, all edge maps used in GVF computations were normalized to the range $[0, 1]$ in order to remove the dependency on absolute image intensity value. The snakes were dynamically reparameterized to maintain contour point separation to within 0.5–1.5 pixels (cf. [11]). The GVF, GGVF and snake parameters are given for each case.

A comparison between the performance of the GVF snake and the GGVF snake is shown in Fig. 1. Using an edge map obtained from the original image shown in Fig. 1(a), both the GVF field ($\mu = 0.2$) and the GGVF field ($K = 0.05$) were computed, as shown zoomed in Fig. 1(b) and 1(c),

respectively. We note that in this experiment both the GVF field and the GGVF field were normalized with respect to their magnitudes and used as external forces. Next, a snake ($\alpha = 0.25$, $\beta = 0$) was initialized at the position shown in Fig. 1(d) and allowed to converge within each of the external force fields. The GVF result, shown in Fig. 1(e), stops well short of convergence to the long, thin, boundary indentation. On the other hand, the GGVF result, shown in Fig. 1(f), is able to converge completely to this same region. It should be noted that both GVF and GGVF have wide capture ranges (which is evident because the initial snake is fairly far away from the object), and they both preserve subjective contours (meaning that they cross the short boundary gaps).

It turns out that a good result similar to that of GGVF in Fig. 1(f) can be achieved using GVF with $\mu = 0.01$. Because μ is small in homogeneous regions as well as near the edges, the convergence of GVF is very slow – it takes an order of magnitude longer than GGVF or GVF with $\mu = 0.2$. If the GVF iterations are terminated early, then the result has an undesirably small capture range. This result shows that GGVF can be thought of as a faster GVF that preserves boundary detail and has a large capture range. The GGVF and GVF results will never be exactly the same, however, since the smoothing parameter of GGVF goes to zero at edges, an impossibility for GVF.

We compared the accuracy of different active contour formulations using the simple harmonic curves. These curves were generated according to the equation

$$r = a + b \cos(m\theta + c)$$

by setting a , b , c to suitable values and varying m . Curves corresponding to $m = 0, 2, 4, 6$ and 8 were digitized on a 201×201 grid to give the images in Fig. 2. In order to eliminate the problem of capture range for traditional active contours so that comparisons could be made, we initialized the active contours at the true curves, and let them deform under the different external forces. After convergence, we computed the maximum distance in the radial direction between the true boundary and each active contour as in [5]. To compute the maximum radial error (MRE), all the final active

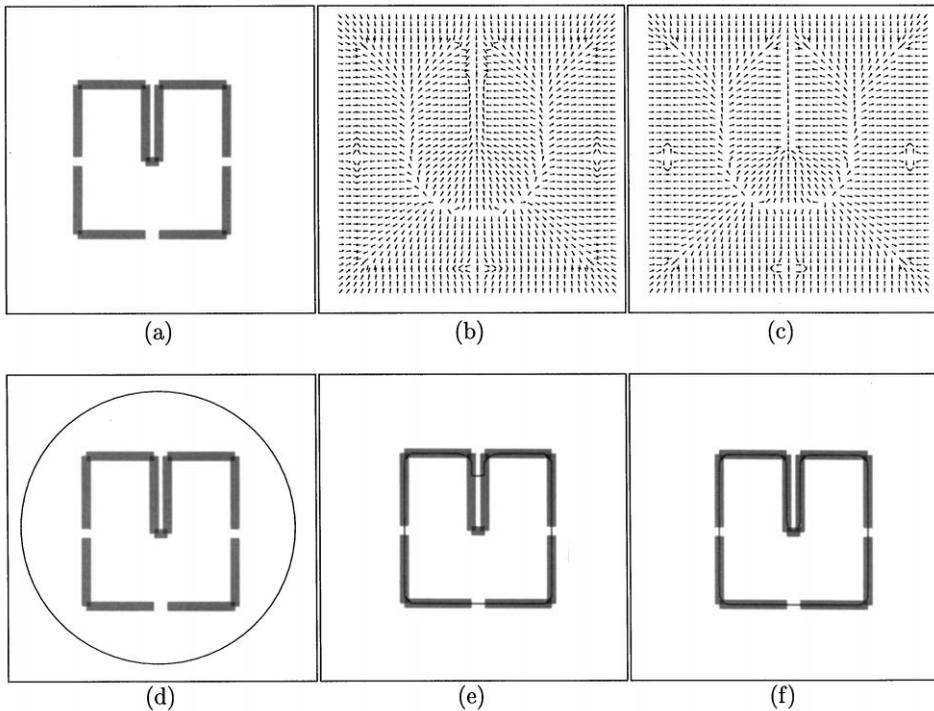


Fig. 1. (a) A square with a long, thin indentation and broken boundary; (b) original GVF field (zoomed); (c) proposed GGVF field (zoomed); (d) initial snake position for both the GVF snake and the GGVF snake; (e) final result of the GVF snake; and (f) final result of the GGVF snake.

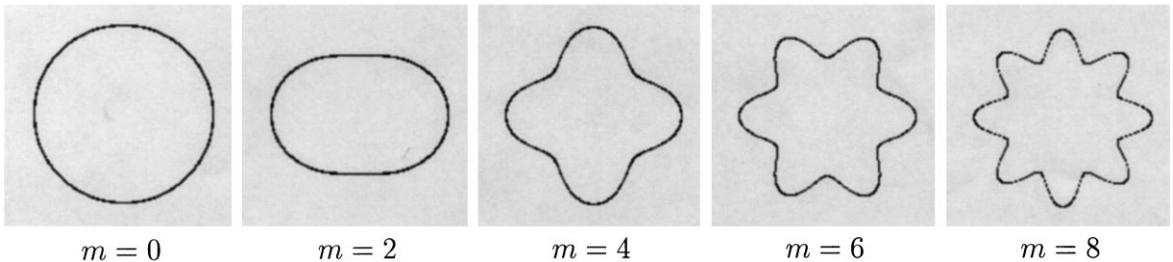


Fig. 2. Harmonic curves: $r = a + b \cos(m\theta + c)$.

contours were linearly interpolated to maintain a pre-specified small point separation. The maximum radial errors were measured in terms of pixels.

Our experimental results on accuracy are shown in Fig. 3. The first three curves shown in this figure resulted from traditional active contour external forces $-\nabla E_{\text{ext}} = -\nabla(G_{\sigma}(x, y) * I(x, y))$ for three Gaussian standard deviations ($\sigma = 1$ pixel, $\sigma = 3$

pixels and $\sigma = 6$ pixels). The fourth curve resulted from the use of the distance potential forces of Cohen and Cohen [4]. The last two resulted from GVF ($\mu = 0.1$) and GGVF ($K = 0.05$). In both cases the test intensity images were used as edge maps. We see that traditional potential forces with small σ yield small errors. Since the capture range of this type of force is very small, however, larger σ 's are

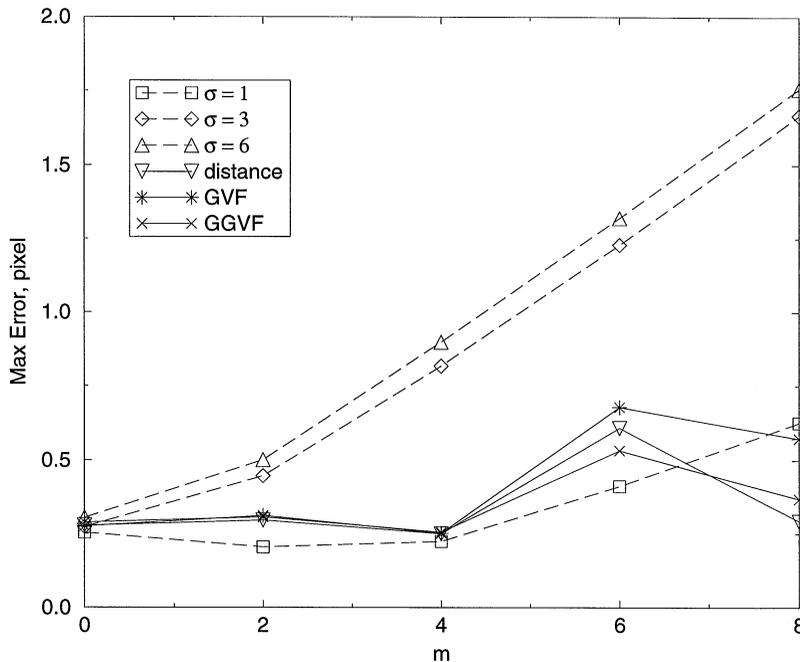


Fig. 3. Maximum radial error (MRE).

often used as the figure shown, these forces do not yield high accuracy, especially at larger m 's. The distance potential forces, GVF forces and GGVF forces, all yield high accuracy consistently. Distance potential forces, however, have been shown to have poor performance on boundary concavities [16]. We note that the fluctuations of the error curves with increasing m arise due to discretization of the curves on the image grid and to the underlying performance variations of active contours.

Active contour algorithms can sometimes be extremely sensitive to noise. To test the noise sensitivity of GVF and GGVF, we added impulse noise to the $m = 8$ harmonic image in Fig. 2. The resulting image is shown in Fig. 4(a) with an initial active contour plotted as a circle. The active contour was then allowed to converge, being driven by external force fields calculated from the noisy image. The results for traditional snakes with $\sigma = 1$ and $\sigma = 9$ are shown in Fig. 4(b) and 4(c), respectively. The problem with Fig. 4(b) is that the snake is simply captured by the local impulsive spikes, rather than the dominant figure. In Fig. 4(c), the large σ blurs

the boundary too much and the snake cannot latch onto the detail. The contour resulting from the distance potential forces is shown in Fig. 4(d). Since this external force uses a binary edge map to begin with, it is attracted to the nearest detected edge points, which do not belong to the dominant figure. The results of GVF and GGVF are shown in Fig. 4(e) and 4(f), respectively. These results, barely distinguishable from each other, demonstrate a remarkable ability to be both captured from a long distance and to converge extremely well to the dominant shape.

It is natural to ask whether there might be a smoothing strategy that would improve the results of the distance potential forces. For example, it may be possible to improve the edge map by prefiltering the image before creating the edge map or by applying a nonlinear filter to the edge map itself. We have tried several approaches along these lines and have found that it is very difficult to eliminate extraneous boundary points while simultaneously preserving the boundary itself. Another approach is to filter the distance potential itself in order to

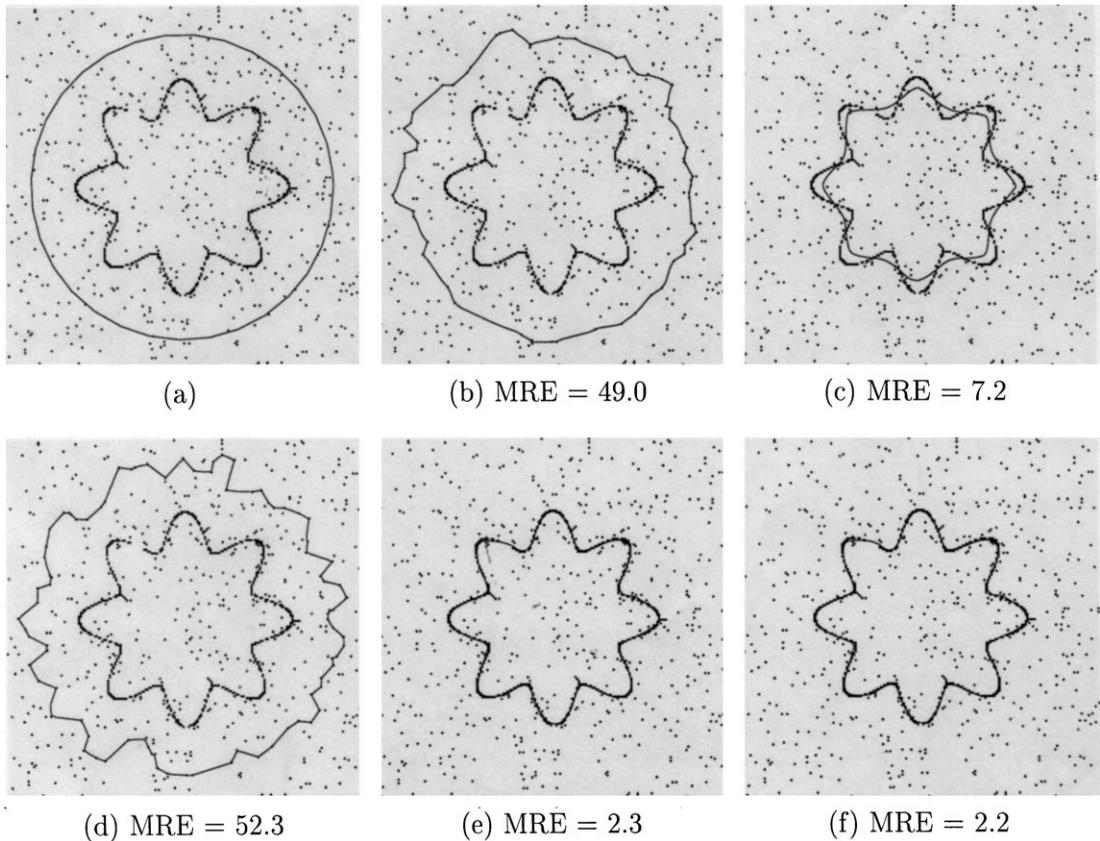


Fig. 4. (a) Impulse noise corrupted image and the initial snake; (b) and (c) snake results using traditional external forces $\nabla(G_\sigma(x, y) * I(x, y))$ where $\sigma = 1$ and 9; (d) snake result using distance potential force; (e) GVF snake result with $\mu = 0.1$; and (f) GGVF snake result with $K = 0.2$. The edge map used for both GVF and GGVF snake is $f = G_\sigma(x, y) * I(x, y)$, where $\sigma = 1$, respectively. All snake results are computed using $\alpha = 0.25$ and $\beta = 0$.

smooth out the energy valleys caused by the extraneous edge points. This approach flattens the valley in which the true edge is located and does not eliminate the extraneous valleys, and the converged active contour has poor fidelity to the truth.

GVF and GGVF both improve over the distance potential forces by applying a very narrow filter to the edge map, followed by a vector diffusion that allows the dominant edge map to obliterate the effects of the extraneous edge points scattered throughout the image. It should be noted that if GVF were run with a small μ parameter, it would not smooth out the extraneous edges. This highlights an important advantage of GGVF over GVF: that GGVF can support convergence to very thin

boundary concavities while simultaneously eliminating extraneous edge points.

Finally, we compared the qualitative performance of GVF and GGVF active contours on a magnetic resonance image of the left ventricle of a human heart. The original image is shown in Fig. 5(a), and its gray-level edge map is shown in Fig. 5(b). The goal in this experiment is to extract the boundary description of the inner wall or endocardium of the left ventricle. The initial positions of both GVF and GGVF active contours are shown as circles in gray overlaid on the real images (Fig. 5(c) and 5(d)). The final contours are shown in white. Many details of the endocardial border are captured by both GVF and GGVF, however, the papillary muscle

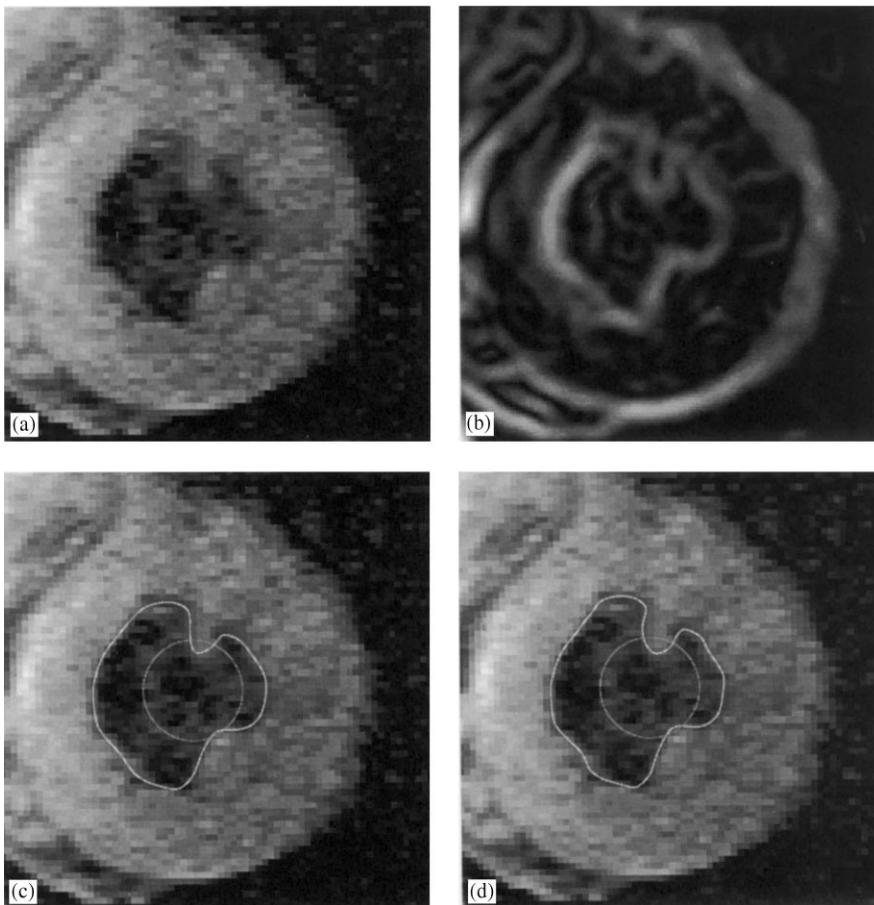


Fig. 5. (a) A 160×160 -pixel magnetic resonance image of the left ventricle of a human heart; (b) the edge map $|\nabla(G_\sigma(x, y) * I(x, y))|^2$ with $\sigma = 2.5$; (c) the result of GVF snake with $\mu = 0.1$; and (d) the result of GGVF snake with $K = 0.15$. The parameters used for both snakes are $\alpha = 0.1$ and $\beta = 0$.

protruding into the cavity at about the 1 o'clock position is represented best by GGVF.

In many cases, GGVF and GVF will perform very similarly. Our experiments have revealed certain differences, however, and these may be important in practice. GGVF will generally show better convergence to thin boundary concavities. If the μ parameter is sufficiently small, however, GVF may achieve similar convergence properties. But in this case, GVF will require significantly longer computation time, and noise in the edge map may cause erroneous convergence. In short, GGVF can be thought of as a computationally faster version of GVF, with better boundary localization, especially

with respect to concave boundaries, and with better noise immunity.

5. Conclusion

We have presented a new class of external force models for active contours. It is a generalization of the GVF formulation that includes two spatially varying weighting functions. We showed that GGVF improves active contour convergence into long, thin boundary indentations, and maintains other desirable properties of GVF, such as an extended capture range. We also showed that

GGVF has excellent performance on noisy and real medical images. Further investigations into the nature and uses of GGVF are warranted. Also, making connections between GGVF with other applications in image processing and computer vision might provide some new insights or even new solutions to existing problems.

Appendix A. Variational framework for generalizing GVF

GVF can also be generalized by starting from the variational formulation proposed in [15]. Spatially varying weighting functions can be used, leading to the following new variational formulation:

$$\varepsilon = \iint g(|\nabla f|) |\nabla \mathbf{v}|^2 + h(|\nabla f|) |\mathbf{v} - \nabla f|^2 \, dx \, dy,$$

where $|\cdot|$ is a vector norm and $\nabla \mathbf{v}$ is second-order tensor. Using the calculus of variations, we obtain the following Euler equation:

$$\nabla \cdot [g(|\nabla f|) \nabla \mathbf{v}] - h(|\nabla f|) (\mathbf{v} - \nabla f) = \mathbf{0}.$$

The solution of this vector equation can be obtained by computing the steady state of the following generalized diffusion equation:

$$\mathbf{v}_t = \nabla \cdot [g(|\nabla f|) \nabla \mathbf{v}] - h(|\nabla f|) (\mathbf{v} - \nabla f),$$

or written more explicitly

$$\mathbf{v}_t = \nabla g(|\nabla f|) \cdot \nabla \mathbf{v} + g(|\nabla f|) \nabla^2 \mathbf{v} - h(|\nabla f|) (\mathbf{v} - \nabla f).$$

This result is different than GGVF. To understand the nature of the difference, we note that if $\nabla g(|\nabla f|) \cdot \nabla \mathbf{v} = \mathbf{0}$, we get GGVF. This condition is data-dependent, however, and is satisfied in homogeneous regions, but is generally nonzero near the edges. We have implemented this generalized GVF and found that it has very similar properties as GGVF and usually yields a very similar result. This version is more computationally demanding, however. Therefore, despite the aesthetically pleasing property that it satisfies a minimum principle, we

advocate GGVF when a generalization to GVF is desired.

References

- [1] A.J. Abrantes, J.S. Marques, A class of constrained clustering algorithms for object boundary extraction, *IEEE Trans. Image Process* 5 (November 1996) 1507–1521.
- [2] I. Carlbom, D. Terzopoulos, K.M. Harris, Computer-assisted registration, segmentation, and 3D reconstruction from images of neuronal tissue sections, *IEEE Trans. Med. Imaging* 13 (June 1994) 351–362.
- [3] L.D. Cohen, On active contour models and balloons, *CVGIP: Image Understanding* 53 (March 1991) 211–218.
- [4] L.D. Cohen, I. Cohen, Finite-element methods for active contour models and balloons for 2-D and 3-D images, *IEEE Trans. Pattern Anal. Machine Intell.* 15 (November 1993) 1131–1147.
- [5] C. Davatzikos, J.L. Prince, An active contour model for mapping the cortex, *IEEE Trans. Med. Imaging* 14 (March 1995) 65–80.
- [6] R. Durikovic, K. Kaneda, H. Yamashita, Dynamic contour: a texture approach and contour operations, *Visual Comput.* 11 (1995) 277–289.
- [7] F.P. Ferrie, J. Lagarde, P. Whaite, Darboux frames, snakes, and super-quadratics: Geometry from the bottom up, *IEEE Trans. Pattern Anal. Machine Intell.* 15 (August 1993) 771–784.
- [8] A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- [9] M. Kass, A. Witkin, D. Terzopoulos, Snakes: Active contour models, *Internat. J. Comput. Vision.* 1 (4) (1987) 321–331.
- [10] F. Leymarie, M.D. Levine, Tracking deformable objects in the plane using an active contour model, *IEEE Trans. Pattern Anal. Machine Intell.* 15 (6) (1993) 617–634.
- [11] S. Lobregt, M.A. Viergever, A discrete dynamic contour model, *IEEE Trans. Medical Imaging* 14 (March 1995) 12–24.
- [12] T. McInerney, D. Terzopoulos, A dynamic finite element surface model for segmentation and tracking in multi-dimensional medical images with application to cardiac 4D image analysis, *Comput. Med. Imaging Graph.* 19 (1) (1995) 69–83.
- [13] D. Terzopoulos, K. Fleischer, Deformable models, *Visual Comput.* 4 (1988) 306–331.
- [14] D. Terzopoulos, R. Szeliski, Tracking with Kalman snakes, in: A. Blake, A. Yuille (Eds.), *Artificial Intelligence*, MIT Press, Cambridge, MA, 1992, pp. 3–20.
- [15] C. Xu, J.L. Prince, Gradient vector flow: A new external force for snakes, *IEEE Proc. Conf. on Comput. Vis. Patt. Recog. (CVPR)* 1997, pp. 66–71.
- [16] C. Xu, J. L. Prince, Snakes, shapes, and gradient vector flow, *IEEE Trans. on Image Process.* (March 1998) 359–369.