IMAGE SEGMENTATION USING CURVE EVOLUTION AND FLOW FIELDS

Baris Sumengen, B. S. Manjunath, and Charles Kenney
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106-9560
{sumengen, manj, kenney}@ece.ucsb.edu

ABSTRACT

An image segmentation scheme that utilizes image-based flow fields in a curve evolution framework is presented. Geometric curve evolution methods require an edge function and a vector field with certain characteristics that are obtained from the image itself. A vector field borrowed from the edgeflow segmentation [2] method is utilized both to obtain an edge function and to guide the curve evolution towards the object boundaries. This vector field is computed from the image using intensity, texture and color features. The proposed method integrates well-tested image features to the well-studied curve evolution methods thus achieving better segmentation results.

1. INTRODUCTION

Image segmentation is an important first step in many image processing and computer vision tasks. It can be defined as a partitioning of the image into homogenous regions. One of the popular approaches to image segmentation is curve evolution and active contour models [1,4,8,11,12,13]. Curve evolution methods usually result in closed contours as opposed to disconnected edges resulting from filtering methods. However, their effectiveness in segmenting natural images that are rich in texture has not been clearly demonstrated. On the other hand some recent filtering-based image segmentation methods have been successfully applied to a variety of images. Example for an edge-based segmentation method is the edgeflow technique [2], which uses a vector diffusion method to find edges. Our aim is to create a combined method that will result a better segmentation that can be applied to a vast variety of images.

The proposed method integrates the edgeflow vector field to the curve evolution framework. The vector field in edgeflow method is calculated by using intensity, texture and color features of the image. This solves the problem of utilizing diverse set of image features with the curve evolution. The commonly used edge function is then generated from the edgeflow vector field by solving a Poisson equation.

The rest of the paper is organized as follows. We review active contour methods and edgeflow methods in section 2. In section 3, we present an edge-based hybrid approach to segmentation using edgeflow and geometric active contours. In section 4 we give some experimental results and conclude with discussions in section 5.

2. PREVIOUS WORK

Active contours and curve evolution methods usually define an initial contour C_0 and deform it towards the object boundary. The problem is formulated using partial differential equations (PDE). The previous research follows two different paths in terms of implementation of active contours, namely parametric active contours (PACs) and geometric active contours (GACs). PACs use a parametric representation of the curves and GACs utilize level set methods [3,6]. Recently some connections between these two methods have been established [1,7]. Our method uses the GAC framework for the implementation of the proposed algorithm.

Curve evolution methods can utilize edge information [1,4,8], regional properties [11,12] or a combination of them [13]. Edge-based active contours try to fit an initial closed contour to an edge function generated from the original image. The edges in this edge function are not connected, so they don't identify regions by themselves. An initial closed contour is slowly modified until it fits on the nearby edges.

Let $C(j):[0,1] \rightarrow \Re^2$ be a parameterization of a 2-D closed curve. A fairly general curve evolution can be written as:

$$\frac{\partial C}{\partial t} = (a + bk)\vec{N} + (\vec{S} \cdot \vec{N})\vec{N} \qquad (1)$$

where k is the curvature of the curve, $\vec{N}$ is the normal vector to the curve, a, b are constants, and $\vec{S}$ is an underlying velocity field whose direction and strength depend on the time and position but not on the curve front itself. This equation will evolve the curve in the normal

direction. The first term is a constant speed parameter that expands or shrinks the curve, second term uses the curvature to make sure that the curve stays smooth at all times and the third term guides the curve according to an independent velocity field.

In their independent and parallel works, Caselles et al. [8] and Malladi et al. [4] initialize a small curve inside one of the object regions and let the curve evolve until it reaches the object boundary. The evolution of the curve is controlled by the local gradient. This can be formulated by modifying (1) as:

$$\frac{\partial C}{\partial t} = (F + e k) g \vec{N} \qquad (2)$$

where F, e are constants, and $g = 1 / \left(1 + \left|\nabla \hat{I}\right|\right)$. $\hat{I}$ is the Gaussian smoothed image. This is a pure geometric approach and the edge function, g, is the only connection to the image. Caselles et al. [1] introduced geodesic active contours, which is an improvement over the previous active contour methods. Starting with the *snakes* problem defined by Kass et al. [5], they reformulated the energy functional as a geodesic computation in a Riemannian space and found the following gradient descend equation:

$$\frac{\partial C}{\partial t} = (F + k) g \vec{N} - (\nabla g \cdot \vec{N}) \vec{N} \qquad (3)$$

Here $-\nabla g$ defines a vector field on the pixels of the image. The corresponding vectors are orthogonal towards the closest boundary or edge. The advantage of this method over the pure geometric approaches is that even if the curve propagates beyond the boundary, the ∇g term in (3) pulls the curve back towards the boundary. Even though an improvement over previous methods, this method is still prone to boundary leaking as shown in [14]. Edgeflow image segmentation [2] is a recently proposed method that is based on filtering and vector diffusion techniques. Its effectiveness has been demonstrated on a large class of images. It features multiscale capabilities and uses multiple image attributes such as texture and color. First, a vector field is defined on the pixels of the image grid (Fig 1b). At each pixel, the vector's direction is oriented towards the closest image discontinuity at a predefined scale. The magnitude of the vectors depends on the strength and the distance of the discontinuity. After generating this vector field, a vector diffusion algorithm is applied to detect the edges. This step is followed by edge linking and region merging to achieve a partitioning of the image. Details can be found in [2].

3. COMBINING EDGEFLOW AND GAC

Most of the previous research on curve evolution emphasize the geometric nature of the problem and doesn't establish close connections to diverse set of characteristics that can be extracted from an image. These image characteristics are very important for the quality of the segmentation results. On the other hand, these PDE-based curve evolution techniques establish a nice theoretical framework to work on. The real challenge is to utilize image attributes effectively for image segmentation. The recently proposed edgeflow method is quite effective on large and diverse class of images, but requires post processing to detect closed contours. One of the contributions of our proposed method is to utilize the effective edgeflow method within the curve evolution framework to obtain better segmentation results.

Two key components shaping the curve evolution are the edge function g and the external force field $\vec{F}_{ext}$. The purpose of the edge function is to stop or slow down the evolving contour when it is close to an edge. So g is defined to be close to 0 on the edges and 1 on homogeneous areas (Fig. 1c). A popular choice of g is $g = 1 / \left(1 + \left|\nabla \hat{I}\right|\right)$. The external force vectors $\vec{F}_{ext}$ attract the active contour towards the boundaries. At each pixel, the force vectors point towards the closest object boundary on the image (Fig. 1b).

We use the edgeflow vector field as our external force field in a GAC formulation. Similar to other external force fields, edgeflow vectors are designed to point towards the closest boundaries. One of the advantages of edgeflow is that it doesn't depend directly on image gradients. It can be adjusted to work at different scales, and it is easily extendible to color and texture images. Even though the PDEs look very similar, this approach has a fundamental difference compared to the GACs. The setup of the GACs requires that an edge function g is designed. From this edge function, the force vectors are automatically generated using the equation $\vec{F}_{ext} = \nabla g$. The success of the segmentation depends on how good the edge function reflects the boundaries on the image. On the other hand, in our proposed method, $\vec{F}_{ext}$ —the edgeflow vector field—is custom designed using diverse set of image features including color and texture. An edge function is then obtained from $\vec{F}_{ext}$.

It has been shown in [7] with comparison to its counterparts that custom designing $\vec{F}_{ext}$ can lead to better results and fix the shortcomings of geometric active contours such as boundary leaking. Only recently this external force field borrowed from PACs is integrated to GACs [9]. One of the shortcomings in the design of both edge functions and external forces is that they depend directly on the image gradients as the boundary locations

even though it has been shown that the image gradient is very sensitive to the noise and is not very reliable.

To be consistent with the formulation of the GACs, which require $\vec{F}_{ext} = \nabla g$, we find a potential function whose gradient is the edgeflow vector field. One problem with this is that there is no guarantee that edgeflow is a conservative field and this potential function exists. To solve this problem, we separate the conservative and non-conservative components of edge flow field. Let $\vec{S}$ be the edgeflow vector field. According to the Helmholtz theorem, any vector field can be written as a sum of an irrotational (conservative) and a solenoidal vector field. So the edge flow vector field can be written as

$$\vec{S} = -\vec{\nabla} V + \vec{\nabla} \times \vec{A} \qquad (4)$$

taking the divergence of both sides

$$\vec{\nabla} \cdot \vec{S} = -\nabla^2 V + \underbrace{\vec{\nabla} \cdot (\vec{\nabla} \times \vec{A})}_{0} \qquad (5)$$

Since the second term is zero, we only need to solve a Poisson equation [15] to find the edge function V

$$\vec{\nabla} \cdot \vec{S} = -\nabla^2 V \qquad (6)$$

Having generated both the edge function and the external force field, our proposed curve evolution equation is

$$\frac{\partial C}{\partial t} = V \boldsymbol{a} \vec{N} + (\vec{S} \cdot \vec{N}) \vec{N} + V \boldsymbol{k} \vec{N} \qquad (7)$$

where $\boldsymbol{a}$ is a constant, $\boldsymbol{k}$ is the curvature, and $\vec{N}$ is the normal to the curve.

Edgeflow vectors can be calculated based on the image features such as pixel intensity, color, texture or combinations of them. Unlike other edge-based active contour methods, applying our segmentation method to texture or color images doesn't require any changes in the formulation of the curve. This is because the flow vector calculation is separated from the curve evolution.

4. EXPERIMENTAL RESULTS

The implementation of the proposed method consists of three steps. In the first step, the edgeflow vectors are generated. Using this vector field, an edge function is obtained by solving a Poisson equation. These outputs are then used in the third step wherein a manually instantiated curve is propagated according to (7). Curve propagation is
then repeated until the curve converges to a stable boundary.

The edgeflow vectors are calculated using a predefined scale parameter. This vector field calculation is conducted using intensity, color, or texture features or a combination of them depending on the type of the image. For a detailed discussion of edgeflow computations we refer to [2]. After calculating the edgeflow vector field $\vec{S}$, the edge function

V is obtained by solving (6) with Neumann boundary conditions. Fig. 2 demonstrates this process on a natural image where a tiger is standing in a grass field. Texture and color features of the image have been used in the calculations. Fig. 2c shows the edgeflow vector field of a part of the image and Fig. 2b is the edge function calculated from this vector field.

After obtaining $\vec{S}$ and V, the well-known level set method formulation [3,6] is used to implement the curve evolution in (8). This requires defining a corresponding level set function U that embeds C as its zero level set and the time evolution of U. The level set equation corresponding to (8) is

$$\frac{\partial U}{\partial t} = V(\boldsymbol{a} + \boldsymbol{k}) |\nabla U| + \vec{S} \cdot \nabla U \qquad (8)$$

Here U is a 3-D function where $U(x, y) = 0$ defines the evolving curve. For more details refer to [6].

The segmentation results on the tiger image are shown in Fig. 3. A curve is instantiated close to the borders of the image in Fig. 3a. This curve then shrinks and captures the boundary of the tiger in Fig 3(b-c). $\boldsymbol{a}$ is set as –1 in (7) to shrink the curve. In Fig. 3d, a small contour is initialized inside the tiger. $\boldsymbol{a}$ is set as 1 to expand the curve (Fig 3e) and the corresponding segmentation result is shown in Fig. 3f.

Fig 4 shows segmentation of a lizard image using multiple independent curves. 24 curves are instantiated at different locations in the image and all of them are expanded. The curve evolutions are guided by color and texture features.

5. DISCUSSION

We have presented a new segmentation method by integrating edgeflow method to the curve evolution. Images are successfully processed using texture and color features. A distinguishing feature of the proposed method is that in this approach the vector field generation is separated from the curve evolution. In the past, the assumption was that the vector field needs to be generated by a PDE. Because of the mathematical complexity of it, mostly simple designs were used. The ability to freely design the vector field independent from the curve evolution facilitates a better analyses of image segmentation. Edgeflow vector field is used in our implementation but any vector field with similar characteristics, such as the one in [10].

11. REFERENCES

[1] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic Active Contours," Int. J. of Comp. Vision, pp. 61-79, Feb.-March 1997.

[2] W. Ma, B.S. Manjunath, "EdgeFlow: a technique for boundary detection and image segmentation," Trans. Image Proc., pp. 1375-88, Aug. 2000.

[3] S. Osher, J. A. Sethian, "Fronts propagating with Curvature dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," J of Computational Physics, 79, pp. 12-49, 1988.

[4] R. Malladi, J.A. Sethian, B.C. Vemuri, "Evolutionary fronts for topology-independent shape modeling and recovery" ECCV'94.

[5] M. Kass, A. Witkin, D. Terzopoulos, "Snakes: active contour models," Int. J. of Comp. Vision, pp. 321-31, 1987.

[6] J. A. Sethian, "Level set methods and fast marching methods", Cambridge University Press, 1999.

[7] C. Xu, A. Yezzi, L. Prince, "On the Relationship between Parametric and Geometric Active Contours", Technical Report JHU/ECE 99-14, Dec 1999.

[8] V. Caselles, F. Catte, T. Coll, F. Dibos, "A geometric model for active contours in image processing", Num. Mathematik, 1993. p.1-31

[9] N. Paragios, O. Mellina-Gottardo, V. Ramesh, "Gradient Vector Flow Fast Geodesic Active Contours", ICCV 2001.

[10] M. Tabb, N. Ahuja, "Multiscale image segmentation by integrated edge and region detection", Trans. on Image Proc., May 1997. p.642-55.

[11] A. Tsai, A. Yezzi, A.S. Willsky, "A curve evolution approach to smoothing and segmentation using the Mumford-Shah functional", CVPR 2000, p.119-24.

[12] T.F. Chan, L.A. Vese, "Active contours without edges", Trans. on Image Proc., Feb. 2001. p.266-77.

[13] N. Paragios, R. Deriche, "Geodesic active regions for supervised texture segmentation", ICCV 1999, p.926-32.

[14] K. Siddiqi, Y.B. Lauziere, A. Tannenbaum, S.W. Zucker, "Area and length minimizing flows for shape segmentation", Trans. Image Proc., March 1998, p.433-43.

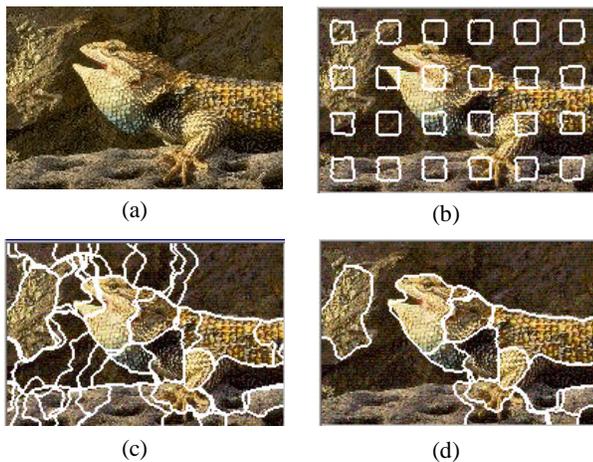[15] An introduction to numerical analysis, Kendall E. Atkinson, John Wiley and Sons, New York, 1989.
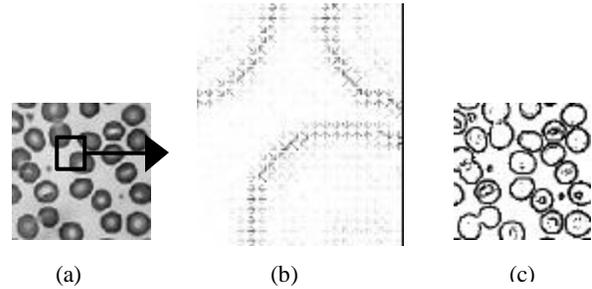
Fig. 1. (a) Image of blood cells. (b) Edgeflow vector field corresponding to the rectangle on the image. (c) An example for an edge function.

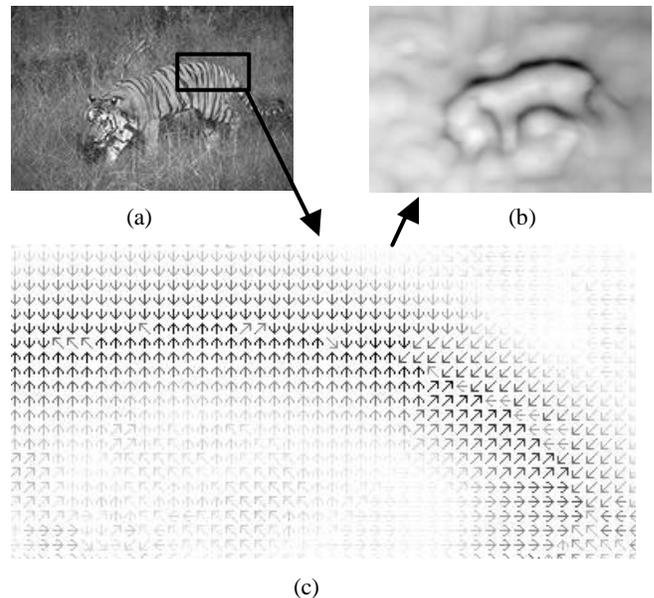Figure 2. (a) An image of a tiger. (b) Edge function. (c) Edgeflow vector field corresponding to the rectangle on the original image.

Figure 4. (a) A lizard image (b) Multiple initial curves are instantiated. (c) Evolving curves. (d) Segmentation result.
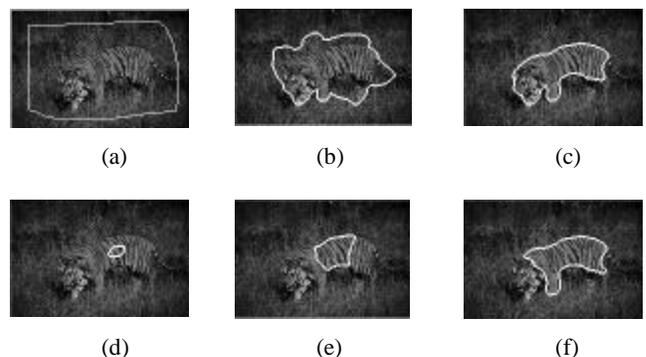

Figure 3. (a) An initial curve is instantiated. (b-c) Corresponding curve evolution. (d) A curve is instantiated inside the tiger. (e-f)