AN IMPROVED SNAKE-BASED METHOD FOR OBJECT CONTOUR DETECTION

Shin-Hyoung Kim and Jong Whan Jang

Department of Information and Communication Engineering, PaiChai University
Daejeon, South Korea
{zeros and jangjw}@pcu.ac.kr

ABSTRACT

In this paper we present a snake-based method for efficiently detecting contours of objects with boundary concavities. The proposed method is composed of two steps. First, the object's boundary is detected using the proposed snake model. Second, snake points are optimized by inserting new points and deleting unnecessary points to better describe the object's boundary. We use the Frenet formula to calculate the binormal vector at snake points and use a regional similarity energy to prevent snake points from converging on foreign edges. Moreover, we use the result to control the direction of movement for snake points near boundary concavities. The proposed algorithm can successfully detect boundary of objects. Experimental results have shown that our algorithm produces more accurate contour detection results than the conventional algorithm.

Index Terms— Contour detection, segmentation, Snakes

1. INTRODUCTION

Detecting the boundary of an object is the basis for many important applications such as machine vision, image and video coding and content-based retrieval systems [1]. In fact, the MPEG-4 standard [2] is an object-based image and video coding algorithm that requires object segmentation in the first stage.

Over the past two decades, various object segmentation schemes have been developed for extracting an object from an ordinary image. The active contour (snake) model [3-8] is one such scheme that has been successfully used in computer vision and object recognition. The snake is an energy-minimizing spline guided by internal forces that preserve its characteristics, and external forces that pull it toward image features such as lines and edges. However, the snake-based boundary detection approach suffers from challenges such as the difficulty in progressing into boundary concavities and the fixation of snake points, which is usually fixed. Cohen's method was proposed to address the concavity problem with pressure forces [4]. This model is to have large external forces far away from the desired boundaries, and can push an active contour into boundary concavities, but when it is too strong *weak* edges will be overwhelmed. Moreover, pressure forces must also be initialized to push out or push in, a condition that mandates careful initialization. Xu and Prince proposed the GVF (gradient vector flow) snake [6] to handle both problems. The GVF method uses a spatial diffusion of the gradient of the edge map of the image instead of using the edge map directly as an external force. Although the method has the advantages of insensitivity to initialization and the ability to move into boundary concavities, it can not handle gourd-shaped concavities. This is due to the concentration of GVF energy in the neck of the gourd. Moreover, GVF snakes require a long computation time for generating the GVF map.

In this paper we present a snake-based method for object segmentation addressing the above challenges through a modified snake model with optimized number of points. We propose a new modified internal energy of the snake model and a regional similarity energy as a component of the external energy to handle the concavity problem and ignore foreign edges, respectively. Optimizing the number of snake points is used to better describe the object's boundary.

This paper is organized as follows. Section 2 covers a background on the snake model. Our proposed method for extracting objects with boundary concavities is presented in Section 3. In Section 4 we show experimental results for evaluating the performance of our method, and the conclusions are given in Section 5.

2. THE SNAKE MODEL

In the discrete formulation of a snake [5], the contour is represented as a set of snake points $v_i = (x_i, y_i)$ for $i = 0,...,M-1$ where x_i and y_i are the x and y coordinates of a snake point, respectively and M is the total number of snake points. Then the energy function for a snake point is typically written in the form $E_{snake}(v_i) = E_{int}(v_i) + E_{ext}(v_i)$. The first component is called the *internal energy* and it concerns contour properties such as curvature and discontinuity. The second component is the *external energy* which is typically defined as the gradient of the image at a snake point.

3. PROPOSED ALGORITHM

Our proposed method modifies the conventional snake model, and introduces a region similarity energy and a mechanism for the insertion and/or deletion of snake points as necessary.

3.1 Proposed new snake energy function

In this section, we describe our special snake energy function. Internal energy is typically expressed in terms of two terms: *continuity energy*, and *curvature energy*. Our modifications to these two terms and to the external energy term will be explained in the following subsections.

3.1.1 Internal energy terms

Continuity energy: The main role of the conventional continuity energy term is to make even spacing between the snake points by minimizing the difference between the average distance and the distance between neighboring snake points. In this arrangement point spacing is globally and uniformly constrained by average distance. This is not suitable for objects with concavities because distance between snake points should be relaxed in boundary concavities. Therefore, in our method, point spacing in boundary concavities is allowed to differ from other parts of the snake. We define the normalized continuity energy as follows:

$$E_{con}(v_i) = \frac{\left\| \|v_{i-1} - v_i\| - \|v_i - v_{i+1}\| \right\|}{con_{\max}} \quad (1)$$

where $con_{\max}$ is the maximum value in the search neighborhood. The proposed continuity energy encourages for spacing between neighboring snake points in concave parts of the object's boundary.

Curvature energy: For curvature energy, the goal is to control the smoothness of the contour between neighboring snake points. The way in which this term is formulated affects the ability of snake points to progress into boundary concavities, and conventional snake models show poor performance in this aspect. In this paper we present a new curvature energy solving the problem based on the Frenet formula. As depicted in Fig. 1(a), let $\Psi : I \to \Re^3$ be a unit-speed curve. Then, $T = \Psi'$ is the unit tangent vector field on Ψ, and $N = T'/\|T'\|$ is the principal normal vector field of Ψ; where $\|T'\|$ is the length of the curvature. The vector field $B = T \times N$ on Ψ is the binormal vector field of Ψ, and the sign of the z-dimension of B is positive if B is upward and is negative if it is downward. Therefore, we consider the sign of the binormal vector. In 2D the sign of the binormal vector $B(v_i)$ can be obtained using the cross product of the two vectors $T(v_i)$ and $N(v_i)$ as follows:

$$B(v_i) = T(v_i) \times N(v_i) = \begin{vmatrix} x_i^T & y_i^T \\ x_i^N & y_i^N \end{vmatrix} = (x_i^T y_i^N - y_i^T x_i^N)\vec{e}_z \quad (2)$$

where (x_i^T, y_i^T) and (x_i^N, y_i^N) are the x and y coordinates of the tangent vector $T(v_i)$ and normal vector $N(v_i)$ at the current point v_i, respectively. $\vec{e}_z$ is a unit vector of the z-component. In the discrete formulation of a snake point, $T(v_i)$ and $N(v_i)$ are defined as follows :

$$T(v_i) \approx v_{i+1} - v_i, \quad N(v_i) \approx v_{i-1} - 2v_i + v_{i+1} \quad (3)$$

In the case that a snake point is outside the object (as in Fig. 1(b, c)), the movement of a snake point can be described as follows:

i) When $B(v_i)$ is negative, as the case in Fig. 1(b), v_i moves in the opposite direction of $N(v_i)$ seeking a location with maximum value of $\|N(v_i)\|$.

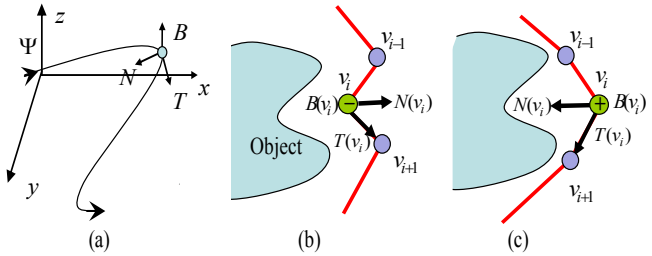


Fig. 1. Movement of snake points based on binormal vector.

ii) When $B(v_i)$ is positive, as the case in Fig. 1(c), the snake point v_i must take the direction of $N(v_i)$ to minimize curvature and to converge on the boundary of the object.

The normalized curvature energy is defined as Eq. (4). It is further weighted by the constant λ, as given in Eq. (5).

$$E_c(v_i) = \left\| T(v_i)/\|T(v_i)\| - T(v_{i-1})/\|T(v_{i-1})\| \right\| \quad (4)$$

$$E_{cur}(v_i) = \lambda E_c(v_i) \quad (5)$$

The constant λ can be set for different applications, relative to the sign of $B(v_i)$ on the contour. In our work, λ is set to $+1$ when the sign of $B(v_i)$ is positive and -1 otherwise.

3.1.2 External energy function

In this subsection, we propose two external energies: *Edge energy*, and *Regional Similarity energy*.

Edge energy is external energy component and is defined as $-|\nabla[G_\sigma(v_i) * f(v_i)]|/e_{\max}$, where $G_\sigma(v_i)$ is a two-dimensional Gaussian function with standard deviation σ and ∇ is the gradient operator. $f(v_i)$ is the image intensity function, and $e_{\max}$ is the maximum value in the search neighborhood.

Regional Similarity energy (RSE) is our proposed energy for pushing the mean intensity of the polygon enclosed by the snake contour to be as close as possible to the mean intensity of the OOI. The smaller intensity difference between the polygon and the object, the closer the snake approaches the object contour. This energy can prevent snake points from converging onto foreign edges in the background.

As illustrated in Fig. 2, the window W_i is searched for the point which produces the minimum polygon area. The contour created by such point is referred to as $C_{\min}$, and is regarded as the base area contour. Mean intensity of the area enclosed by $C_{\min}$ is regarded as the base mean intensity m_i. Next, the window W_i is searched for a point p which yields a polygon with the minimum intensity variance relative to $C_{\min}$. For each candidate point in W_i, a second contour (C_{area}) is produced and the marginal area $C_{area} - C_{\min}$ is yielded. The search seeks a point p that yields a marginal area with minimum variance to $C_{\min}$. We refer to this measure as RSE and is defined as:

$$E_{RSE}(p)=\frac{1}{A}\left(\sum_{u}\sum_{v}\left|I(u,v)-m_i\right|^2\right)\frac{1}{RSE_{\max}},\quad (u,v)\in C_{area}-C_{\min} \qquad (6)$$

where $RSE_{\max}$ is the maximum possible RSE values in the neighborhood, and A is the area of ($C_{area}-C_{\min}$), and can be calculated by using the equation:

$$A=\frac{1}{2}\sum_{i=1}^{n}\begin{vmatrix} x_i & y_i \\ x_{i+1} & y_{i+1}\end{vmatrix} \qquad (7)$$

where (x_{n+1},y_{n+1}) is equal to (x_1,y_1) for a closed snake contour and n is the total number of snake points. In our case, we only consider four snake points (x_{i-1},y_{i-1}), (x_i,y_i), (x_{i+1},y_{i+1}), $(x_{\min},y_{\min})$.

3.2 Optimizing the number of snake points

After snake points converge in the first step to the boundary of the object, additional points are inserted and unnecessary points are deleted to better describe the boundary concavities. The newly inserted point is defined as $c_i=(v_i+v_{i+1})/2$. This method was explained detail in [8].

3.3 Movement of inserted points

Inserted points will not usually fall on the object's boundary and therefore need to be relocated to the boundary. In stereo-based approach, we can find out where an inserted point is located using disparity information [7]. However, in 2D we need a different solution. Fig. 3 illustrates the decision controlling the movement of inserted points taking into consideration whether the inserted point is inside or outside the object. As shown in the Fig. 3, the sign of $B(c_i)$ is negative in concave segments and positive in convex segments. To decide the movement of an inserted point, we first check the sign of its binormal vector $B(c_i)$ relative to previous and next inserted points. If the sign of $B(c_i)$ is negative, then the point is outside, and in that case we apply exactly the same scheme explained in subsection 3.1.1 above. On the other hand, if the inserted point turns out to be inside (i.e., if the sign of $B(c_i)$ is positive), we apply the same scheme except that we reverse the signs for λ.

4. EXPERIMENTAL RESULTS

To verify the performance of our algorithm a set of experiments has been performed.

The algorithm was coded in Visual C++ 6.0, and the experiment was performed on a Pentium IV machine with 1GByte of memory running at 3GHz clock speed. We used binary and color real images with 320×240 and 256×256 image size.

The criterion used to verify the accuracy of the estimated snake region R_{est}, compared with the original object region R_{ori} is *Relative Shape Distortion*, $RSD(R)$, which is defined as:

$$RSD(R)=\left(\sum_{(x,y)\in f}R_{ori}(x,y)\oplus R_{est}(x,y)\right)\Big/\sum_{(x,y)\in f}R_{ori}(x,y) \qquad (8)$$

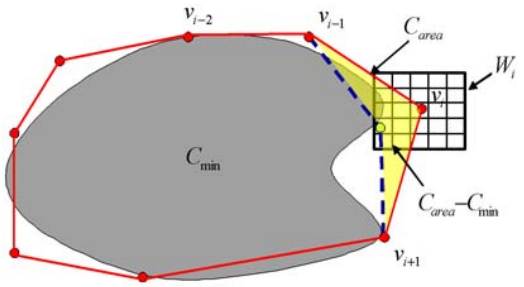


Fig. 2. Regional similarity energy.

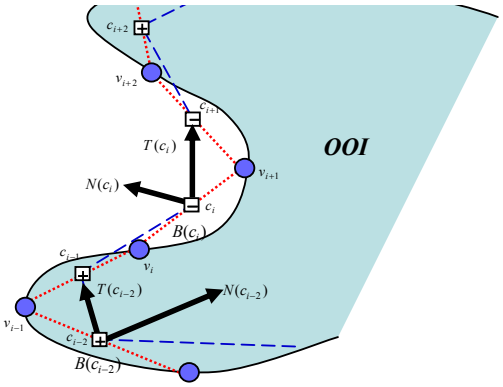


Fig. 3. Movement of inserted points in convex/concave segments of the object's boundary.

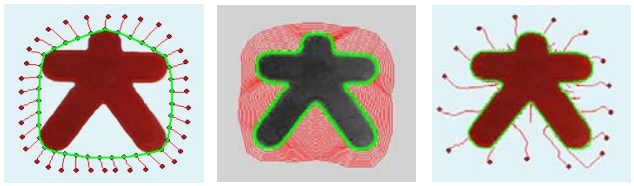
where the $\oplus$ sign is the binary XOR operation. The simulation process and its results are illustrated in Figs. 4 ~ 7, and the performance comparison in terms of *RSD* and computational time is listed in Table 1.

Fig. 4 shows results of an experiment on an ordinary image for an object with boundary concavities. The greedy snake could not detect the boundary of the object, Fig. 4(a), but the GVF snake points converged successfully, Fig. 4(b) because the object has only concavities without gourd-shaped boundary. With our proposed algorithm the snake points also converged onto the boundary concavity in only four optimization iterations, Fig. 4(c).

Fig. 5 and Fig. 6 show results of an experiment on an ordinary and binary image for an object with gourd-shaped concavities. The greedy snake could not detect the boundary concavities of the object as shown in Figs. 5(a) and 6(a), and the GVF snake points failed to proceed beyond the neck of the gourd as shown in Figs. 5(b) and 6(b). However, with our proposed algorithm the snake points converged better onto the boundary concavity in only four optimization iterations as shown in Figs. 5(c) and 6(c). Fig. 6(d) shows the results for our proposed algorithm on an ordinary image.

We performed an additional test involving our algorithm and the GVF snake on an MR image of the left ventricle of a human heart as shown in Fig. 7. Both snakes were initialized inside of the object (ventricle). Our algorithm performed a complete segmentation (b), while the GVF segmented the left side only. The GVF snake can segment the whole ventricle only if it were initialized in the space between the papillary muscles (the bumps that protrude into the cavity).

Performance comparison between the proposed algorithm and the conventional algorithm in terms of *RSD* and computational time is summarized in Table 1.

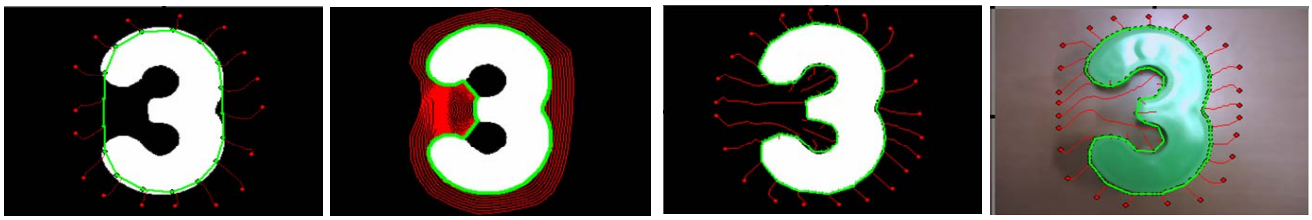


(a) (b) (c)

Fig. 4. Results of experiment on the object with concavities. (a) Greedy snake, (b) GVF snake, (c) Proposed algorithm.

(a) (b) (c)

Fig. 5. Results of experiment on an object with gourd-shaped concavities. (a) Greedy snake, (b) GVF snake, (c) Proposed algorithm.



(a) (b) (c) (d)

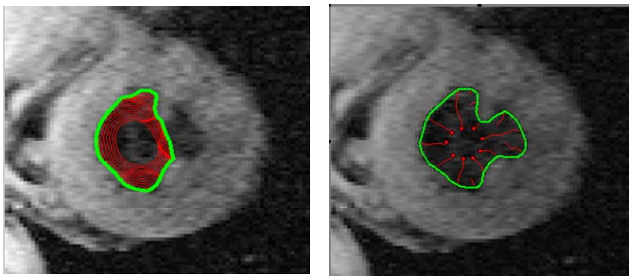
Fig. 6. Results of experiment on the gourd-shaped object: (a) Greedy snake, (b) GVF snake, (c),(d) Proposed algorithm.

Experimental results show that the proposed method gives more accurate results than other methods. Moreover, the computational time of our proposed method is satisfactory, while the GVF snake needs a long time for computation.

5. CONCLUSIONS

In this paper, we have presented a new snake-based scheme for efficiently detecting contours of objects with boundary concavities. Our method extends and improves the conventional snake model. The developed method was tested and showed successful results in extracting objects with boundary concavities.

In the proposed new snake method the movement of snake points is determined using the sign of the cross product of the tangent and normal vectors (i.e. binormal vector). In addition, we proposed a regional similarity energy to detect the boundary and ignore foreign edges. We used optimizing the number of snake points to better describe the object's boundary. In consequence, we can solve the problem which occurs with gourd-shaped boundary concavities. Performance has been evaluated by running a set of experiments using 2D image data having objects with varying degrees of boundary concavity. When compared with conventional snake algorithms, our method has shown a superior object segmentation capability in terms of accuracy. Further research work is being considered to follow up from object segmentation to object tracking in video sequences.



(a) (b)

Fig. 7. Results of experiment on the image of human heart ($\sigma = 2.5$).

Table 1. Performance comparison in terms of *RSD* and time.

		(a) Greedy snake	(b) GVF snake	(c) Proposed algorithm
Fig. 4	RSD (pixel)	(7339/13107) 0.559	(0/13107) 0.000	(12/13107) 0.000
	Time (sec.)	0.048	18.397	0.183
Fig. 5	RSD	(3124/7932) 0.394	(719/7932) 0.091	(100/7932) 0.013
	Time	0.052	12.581	0.096
Fig. 6	RSD	(5839/15143) 0.385	(3880/15143) 0.256	(128/15143) 0.008
	Time	0.073	16.214	0.109

6. REFERENCES

[1] C. L. Huang, D. H. Huang, "A Content-Based Image Retrieval System," Image Vision Comput. vol. 16, pp. 149-163, 1998.

[2] ISO/IEC JTC/SC29/WG11/W4350: "Information Technology - Coding of Audio-Visual Objects Part2: Visual" *ISO/IEC 14496-2,* July 2001.

[3] M. Kass, A. Witkin, and D. Terzopoulos, Snake: Active Contour Models," *Int'l J. Computer Vision*, vol. 1, no. 4, pp. 321-331, 1987.

[4] L. D. Cohen, "On Active Contour Models and balloons," CVGIP: Image Understand., vol. 53, pp. 211-218, Mar. 1991.

[5] D. J. Williams and M. Shah, "A Fast Algorithm for Active Contours And Curvature Estimation," *Computer Vision, Graphics, and Image Processing,* vol. 55, pp. 14-26, 1992.

[6] C. Xu and J. L. Prince, "Snakes, Shapes, and Gradient Vector Flow," *IEEE Trans. Image Processing,* vol. 7, No. 3, pp. 359-369, Mar. 1998.

[7] S. H. Kim and J. W. Jang, "Object Contour Tracking Using Snakes in Stereo Image Sequences," *ICIP 2005*, vol 2, pp. 414-417, Sept. 2005.

[8] S. H. Kim, A. Alattar and J. W. Jang, "Snake-Based Objects Tracking in Stereo Sequences with the Optimization of the Number of Snake Points," *ICIP 2006*, pp. 193-196, Oct. 2006.