Front. Electr. Electron. Eng. China (2006) 2: 141–145
DOI 10.1007/s 11460-006-0005-4

RESEARCH ARTICLE

ZHAO Bao-jun, LI Dong

Improved Snake algorithm for complex target's boundary detection

Abstract The traditional Snake algorithm cannot effectively detect the object edge of an image with non-convex shapes or low SNR. This paper studies the characteristics of this type of image with complex shape target or noise and presents an improved Snake algorithm. The traditional Snake function model and operation strategy are improved by increasing new control energy functions, and the influencing weight of these energy factors is discussed. At the same time, a dynamic arrangement for the Snake points is used to adapt different target shapes. The simulation results indicate that the new Snake model greatly decreases the dependence on the Snake point's initial position and effectively overcomes noise influence. This method enhances the Snake algorithm's ability of detecting object edge.

Keywords Snake model, Image segmentation, Energy function, Edge detection

1 Introduction

Image segmentation is one of the difficult problems in computer vision research. The traditional edge detection algorithm cannot effectively distill the given objects in the image [1]. Kass et al. presented an active contour model (Snake) algorithm based on energy function, and this algorithm is based on the minimum segmentation of the image's unitary energy, which can exactly segment the expected objects.

Aimed at the minimum energy process, many papers presented different algorithms [2–4] and improved the classical Snake algorithm. There are several main problems in the classical Snake algorithm: 1) its detecting capability depends on the profile's initial position of the Snake point and weight adjusting; 2) it cannot distribute these Snake points according to the object's shape automatically, so the object's edge wouldn't be detected better; 3) anti-noise performance is bad, and the problem that the Snake points usually converge the noise points can't be resolved effectively. To solve these problems, this paper presents a new algorithm of Snake point movement and studies the construction of energy function and Snake points' auto-distribution, which allows the improved Snake algorithm to adapt the complicated image's object segmentation better.

2 The original model of Snake algorithm

Classical Snake model can be depicted as a parameterized curve [5]. Supposing $v(s)=(x(s),y(s))$ is outline of the curve, and s is the arc length, then the whole energy of Snake can be expressed as

$$E_S = E_{\text{int}} + E_{\text{ext}} \tag{1}$$

where E_{int} is internal energy and E_{ext} is external energy. Inner energy E_{int} can be usually calculated as:

$$\begin{aligned} E_{\text{int}} &= E_{\text{e}} + E_{\text{b}} \\ &= \frac{1}{2}\int_s \alpha(s)\,|v_s|^2\,\mathrm{d}s + \frac{1}{2}\int_s \beta(s)\,|v_{ss}|^2\,\mathrm{d}s \end{aligned} \tag{2}$$

In the formula, v_s is the first derivative of the parameterized curve, v_{ss} is the second derivative of the parameterized curve, E_{e} is the elasticity energy of the parameterized curve, and E_{b} is the bend energy of the parameterized curve.

External energy E_{ext} determines that whether the outline converges to the image's characteristic points or not, for gray image $I(x,y)$, it can be calculated as:

$$E_{\text{ext}} = -\left|\nabla I(x,y)\right|^2 \tag{3}$$

According to the Snake algorithm, E_s reaches its minimum only when the object's outline curve converges to

---

Translated from *Transaction of Beijing Institute of Technology*, 2004, 24(2):162-165 (in Chinese)

ZHAO Bao-jun(✉), LI Dong
Department of electronic engineering, Beijing Institute of Technology, Beijing 100081, China
E-mail: zbj@bit.edu.cn

its edge, that is to say, the process to minimize E_s realizes the process to detect the object's edge.

It is known that, when the object has a cupped region, intramural shrinking power can not only make the Snake point close up the object's boundary but also makes the Snake point move away from the object's boundary. At the same time, the external energy depends on image gray to produce driving power, so it will drive Snake points to converge to some noise.

3 Improvement of the algorithm

This paper gives three improved points about the application of the Snake algorithm for complex images.

3.1 The improvement of energy function

Firstly, the new improved internal energy function is described as follows:

$$E_{\text{int}}(i) = \alpha E_{\text{d}} + \beta E_{\text{c}} = \alpha K |v_i - P_{\text{c}}|^2 + \beta |v_{i-1} - 2v_i + v_{i+1}|^2 \tag{4}$$

where E_{d} is the drawing energy from object's center point to Snake points, E_{c} is secondary derivative of the outline, K is a constant, v_{i-1} , v_i , v_{i+1} are three adjacent Snake points, and P_{c} is the center point of target estimated by Snake points.

The advantage of this function is that it indirectly extends the search range of Snake points. The first center point is selected, and subsequent center points in the later iterative process are determined through the estimation of each Snake point. There are two stages in this iterative process:

1) In the initiatory stage of the Snake's movement (first ten to fifteen iterative calculation), error is comparatively big when using Snake points to estimate the center point because of the great deviation between the beginning position of the Snake points and the actual outline of the object. Initial position of the center can remain unchanged.

2) After some iteration, each Snake outline approaches the object outline, and the movement of the center point depending on the Snake points' distribution is described as:

$$C = \frac{1}{N}\sum_{i=1}^{N} S_i \tag{5}$$

where S_i is the position of Snake points. In order to prevent center point to move out of the target, the center point should be controlled by the edge's gradient and satisfy

$$E_{\text{cp}} = \max\,(mE_{\text{cd}} + \frac{n}{E_{\text{cg}}}) \tag{6}$$

where E_{cp} is the energy of center point, E_{cd} is the drawing energy of Snake to center point, E_{cg} is the gradient energy of center point, *m* and *n* are coefficients.

Secondly, a new kind of control energy E_{cont} is presented and expressed as:

$$E_{\text{cont}} = \begin{cases} \left|\bar{D}_i - |v_i - v_{i-1}|\right| & i \text{ is even} \\ 0 & i \text{ is odd} \end{cases} \tag{7}$$

In this equation, $\bar{D}_i = (|v_i - v_{i-1}| + |v_{i+1} - v_i|)/2$ is the mean distance of three adjacent Snake points v_{i-1} , v_i , v_{i+1} . The effect of E_{cont} is to make the distribution of Snake points uniform in the local region. If the distance between Snake points v_i and v_{i-1} is less than the mean distance of v_{i-1} , v_i , v_{i+1} , then Snake point v_i trends to v_i' and v_i'' as shown in Fig. 1. In this way, E_{cont} makes Snake points move to the border of large curvature at the same time. It makes the distribution of the whole Snake points outline more reasonable and symmetrical in the local region.

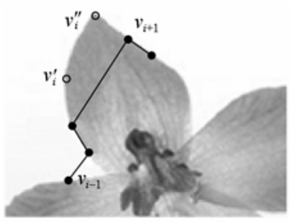

Fig. 1 Explanation of E_{cont}

The energy function in this paper can be described as:

$$E_{\text{S}}(i) = \alpha E_{\text{d}} + \beta E_{\text{c}} + \gamma E_{\text{cont}} + \mu E_i \tag{8}$$

where E_i is the image's energy, $E_i = -|\nabla I(x,y)|^2$, α, β, γ, μ are the weights of every kind of energy respectively.

3.2 Distributing the snake points dynamically

The movement and distribution of Snake points is the core of the Snake algorithm. In order to express the object's outline more accurately and effectively, adaptive control of the quantity and distribution of Snake points is necessary during the movement of Snake points.

Because the edge of the object is fitted by using discrete Snake points, the outline of the object won't be exactly distilled when Snake points are fewer and move to some regions with big curvature. To solve this problem, this paper adopts a strategy that Snake points will be dynamically added according to the magnitude of curvature.

As shown in Fig. 2, v_{i-1} , v_i , v_{i+1} are three neighboring Snake points. The curvature is calculated as Eq. (9). In this formula, $E_i = v_{i-1} - v_i$ and $E_{i+1} = v_{i+1} - v_i$, μ is the cosine of the vertex. When μ becomes bigger, the curvature will become smaller because cosine is a decreasing function in the range $[0, \pi]$. Giving a minimum value $L_{\min}$ and calculating $L = \max(|E_i|,\ |E_{i+1}|)$, if $L \geq L_{\min}$, new Snake point will be added in the midpoint of the corresponding sideline.

$$\mu = \frac{E_i E_{i+1}}{|E_i||E_{i+1}|} \tag{9}$$

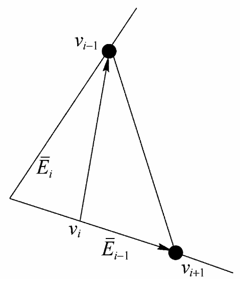

Fig. 2 Dynamic increase of Snake points

On the other hand, considering that some Snake points in the region with smaller curvature have less influence to the whole outline, they can be removed, then the calculation will be greatly reduced. At the same time, to get a better outline, a reasonable value should be set up when removing Snake points to ensure that the distance of two neighboring Snake points is not bigger than the value.

From the method above, we can know that E_{cont} could assure the uniform distribution of Snake points if some points are added or removed and would not disturb other Snake points' distribution.

3.3 Adjustment of the weighted parameter

There are four kinds of energy that act on Snake points in Eq. (8), and they have different effects on the Snake points' movement: E_{d} makes the outline of Snake points move toward the center; E_{c} smoothes the outline; E_{cont} averages Snake points partly; E_i converges Snake points to the true edge of object. During the movement of Snake points, how the four weights are adjusted will greatly affect the final effect of detection.

Analyzing these energies, we can find that: 1) E_{d} and E_{c} in E_{int} should act as the main energy when the Snake point is far from the edge of object. Then, Snake points can move fast to the object and keep away from the background noise in the beginning stage. 2) When the Snake points converge to the edge of the object, E_{c} , E_{cont} and E_i should be the main control energy. These three will drive the Snake points to converge to the edge better, which can decrease the influence of E_{d} greatly after the Snake points have stepped into the outline of the object and avoid missing the control of E_i , then finally converge to the center. Therefore, the key for adjusting the weight is to determine the location relationship between the Snake point and the object's edge, namely, where the Snake points locate: inside, outside or near the outline of object?

The methods below can be used to improve the veracity of estimation.

Determine if the direction D_1 of E_{d} and gradient direction D_2 satisfy the following condition:

$$|D_1 - D_2| < \frac{\pi}{2} \tag{10}$$

Fig.3 shows the estimation method, v_i is Snake point i , and P_{c} is the center of object. It is obvious that the direction of E_{d} always points to the center and the gradient direction is vertical to the edge. The action direction (D_1, D_2) of two energies must be in a range and satisfy the condition Eq. (10) because the center point and Snake point are located on different sides of the edge. This method can filter about half of the background noise (even noise).

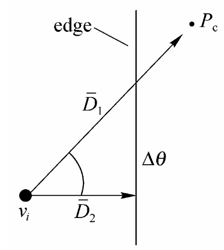

Fig. 3 Edge detection 1

When condition Eq. (10) is satisfied, gray information of the image and Sobel operator can be used to further estimate the object's edge.

For the discrete digital image, the direction of the image's edge is mainly horizontal, vertical or diagonal. Considering the symmetry, there are two cases as shown in Fig. 4. When calculating E_i , the direction of maximum E_i in 8 adjacent regions maybe the edge direction. As shown in Fig. 4, calculating the mean value of gray in area 1 and area 2 respectively (expressed by g_1, g_2) and giving a threshold value T_{g} , if the result satisfies condition Eq. (11), it can be said that Snake point approaches the edge in image gray.

$$|g_1 - g_2| \geq T_{\text{g}} \tag{11}$$

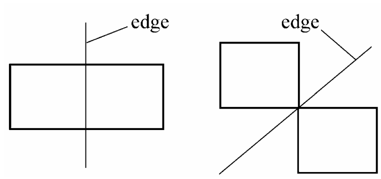

Fig. 4 Edge detection 2

In order to estimate the edge more accurately, we can use Sobel operators again to calculate if the Snake points have a border in the direction of E_i energy (the direction of gradient). If a Snake point meets Eq. (11) and has a border in the direction of E_i according to Sobel operators, then it can be said that this Snake point approached the object's edge. According to the analysis above, weight can be adjusted as follows:

1) When the distance between the Snake point and the object's edge is far, the action effect of E_d is definitive, in this way the noise's effect can be avoided and the object's edge can be approached fast, here $E_{int}(i) = \alpha E_d + \gamma E_c (\alpha > \gamma)$,

2) When the Snake point reaches the target's edge, the coefficient α of E_d is fixed as zero, that is to say, the effect of E_d is canceled. Here, $E_{int}(i) = \beta E_c + \gamma E_{cont} + \mu E_i$.

3) After a Snake point is determined to have reached the object's edge, it's not necessary to find the edge of the object again, which can avoid the Snake point from converging to the center within the object under the influence of E_d .

4) In case two neighboring Snake points are both found to be near the object's edge, a new Snake point is added between them, then the new point is also considered near the object's edge.

4 Experiment result and analysis

In order to validate this algorithm, three different images are used to contrast these results and experiment. Parameters are selected as follows: $\alpha = 1, \beta = \mu = 0,\ \gamma = 0.5$ (not near the object's edge), $\alpha = 0.2$, $\beta = \gamma = 1$, $\mu = 2$ (near the object's edge). The results are shown in Fig. 5, from the top to bottom, they are the pictures plotted by computer, real flowers, flowers with noise in turn.

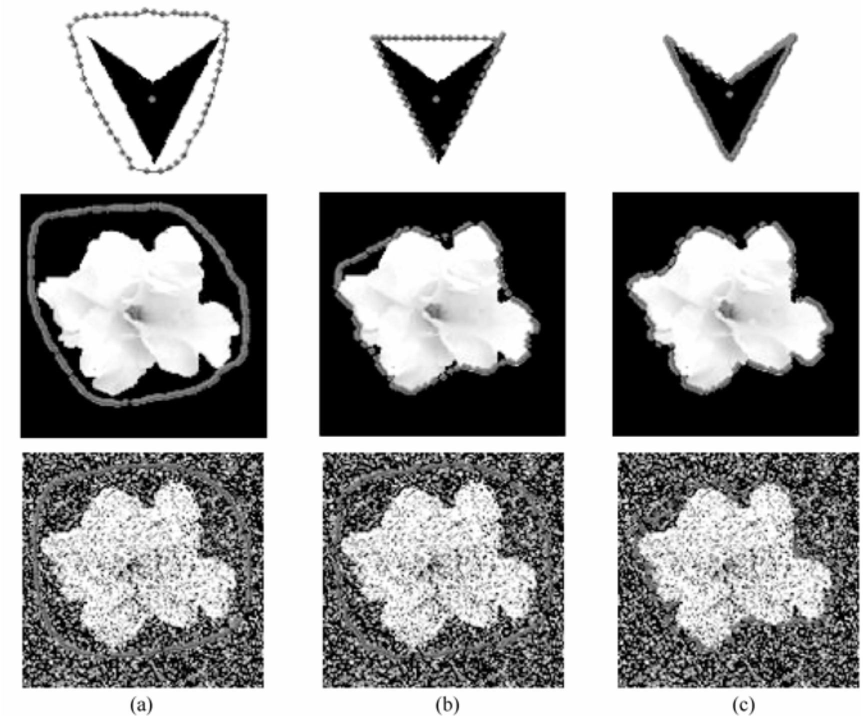

Fig. 5 Simulation result. (a) Initial position of Snake points; (b) Traditional Snake model; (c) New Snake model

According to the simulation results, it is obvious that the iterative results obtained by the new algorithm do not depend on the initial outline's location of Snake point very much. A satisfying edge can be acquired for those images with sunken area and noise. Especially for the images with noise, the classic Snake method can hardly achieve effective iteration, but this new method can detect the flower's edge.

This algorithm emphasizes to resolve the problem that sunken area and noise exist in the images. According to this algorithm, during the iterative process of minimizing Snake points, the location of the Snake point must be determined effectively. Because of the noise effect, it is usually very difficult to make such determination, which greatly affects the iterative results. Some scholars believe that it would be a good research direction to use neural network method to obtain the images' outline in the future [6].

Acknowledgements This study is supported by The National High Technology Research and Development Program of China (863 Program, 2002AA813032).

References

1. Li Hongsong, Hou Chaozhen, Pan Xiuqin, Edge-detection based on the first order absolute central moment, Journal of Beijing Institute of Technology, 2002, 22(3): 279–282
2. Kim Won, Lee Choon-Young, Lee Ju-Jang, Tracking moving object using Snake's jump based on image flow, Mechatronics,

2001, 11: 199–226
3. Cohen L. D., Cohen I., A finite element method applied to new active contour models and 3D reconstruction from cross sections, 3th Inter Conference on Computer Vision. Paris, 1990
4. Deng Hang, Rui Yu, Shi Jun, Snake model for edge detection, Journal of Shanghai Jiaotong University, 2000, 34(6): 848–850
5. Kass M., Witkin M., Terzopoulos D., Snake's: Active contour models, First International Conference of Computer Vision. London，1987
6. Chiou G. I., Hwang Jeng-Neng, A neural network based stochastic active contour model (NNS-Snake) for contour finding of distinct feature, IEEE Trans Image Processing, 1995, 10: 1407–1416