

# Active Contour Model for the Detection of Sharp Corners in Image Boundaries

U. A. A. Niroshika and R. G. N. Meegama

**Abstract** — Active contours are a form of curves that deform according to an energy minimizing function and are widely used in computer vision and image processing applications to extract features of interests from raw images acquired from an image capturing device. One of the major limitations in an active contour is its inability to converge accurately when the object of interest exhibits sharp corners. In this paper, a new technique to deform the active contour by incorporating prior knowledge of significant corners of the object detected using the Harris operator is introduced. The newly constructed active contour deforms towards the boundary of the object without bypassing sharp corner points. Experimental results with several synthetic images show the ability of the new technique to extract features of interest from images consisting of sharp corners with a high accuracy than traditional methods.

**Keywords** — Active Contours, Image Segmentation, Harris Operator, Snakes

## I. INTRODUCTION

Deformable active contour models are widely used in computer vision in image segmentation and boundary detection. It has become more popular in extracting useful information from medical images, especially in medical image analysis that involves Computer Tomography (CT), Magnetic Resonance Imaging (MRI), X-ray and ultrasound images [1, 2].

Extracting boundary elements belonging to the same structure and assimilating these elements into a rational and consistent model of structures is a significant challenge in medical imaging [3]. However, this segmentation task, where the contours of the structures of interest are determined, is both difficult and time consuming to perform manually. Therefore, in order to analyze large volumes of medical image data effectively, high-throughput automated tools are required. Unfortunately, fully automated segmentation techniques fail in producing confident results on biological images due to several issues such as high noise level, low local contrast and numerous structures surrounding the object of interest [4].

This expresses the fact that semi-automated tools and techniques are hence required to segment medical images with high accuracy and with more flexibility. In the past few decades, many effective algorithms have been developed under three main categories namely, thresholding, pattern recognition and deformable models to perform the computer assisted segmentation.

Deformable models can be classified into parametric and geometric models according to the way the contour is depicted. Parametric deformable models represent the curves and surfaces explicitly in their parameter forms during deformation. There are two types of formulations for parametric deformable models: the energy minimizing formulation and dynamic force formulation.

Among these two formulations, the energy minimizing formulation has the benefit that its solution satisfies a minimum principle while the dynamic force formulation has flexibility of permitting the use of more general types of external forces.

The concept of parametric deformable contours, known as snakes, was first introduced by Kass [5] and over the past two decades, it has been enhanced by many researchers worldwide. The original snake has several limitations on its performance. Thus, several new ideas such as: discrete snake, topological adaptive, balloon, fast greedy, gradient vector flow, B-spline, and NURBS have been proposed and added to the concept of the original snake model.

Deformable models and active contour based segmentation have gained ground within computer vision community during the past decade [6]. However, problems such as poor capture range, initial contour placed far from the desired object, problems with concavities, high user interaction and issues with sharp corners, have not been addressed fully in the literature. This research addresses the problem associated with missing corner points near the sharp corners of a desired object.

In our proposed technique, the first stage involves checking the corner points using the Harris operator [7] and detecting them as energy minimization points. Then, the initial snake is reconstructed by adding the detected corner points to capture the boundary. Experiments indicate that the new technique can improve the snake's precision to capture the boundary having sharp corners.

## II. RECENT WORK

The first parametric model known as a “snake” was first proposed by Kass in which the model could only attract the snake towards the boundary when it is initialized near the desired object, or else it may converge to an undesired location. For instance, if the initial snake is placed far away from the subject boundary, it might deform randomly due to lack of gradient force, or it might get attracted towards undesired edges, lines, or noise areas. Consequently, it tends to be difficult for this model to progress into boundary concavities. In order to increase the efficiency and performance of the traditional snake,

U.A.A. Niroshika is with the Department of Information Technology, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka (e-mail: aruni.n@sliit.lk).

R.G.N. Meegama is with the Department of Statistics and Computer Science, Faculty of Applied Science, University of Sri Jayewardenepura, Nugegoda, Sri Lanka (e-mail: rgn@sjp.ac.lk).

numerous new energy terms were added to the energy function later.

A novel efficient model called topology adaptive snake (T-snake) was proposed by McInerny and Terzopoulos [8], by introducing topological flexibility among other features. After every deformation step, the boundary of the object can be determined unambiguously by keeping track of the inside grid vertices. This model has the ability to segment some of the most geometrically complex objects in an efficient and highly automated manner.

Cohen [9] introduced a new model based on pressure forces also known as “balloons”. The pressure force is capable of either inflating or deflating the model. Therefore, in this new model, the initial contour can be placed either inside or outside of the targeted object. This model eliminates the requirement to initialize the contour near the desired object boundary and is also capable of pushing the contour into boundary concavities. Therefore it is more robust to the initial position than the traditional snake. This model has several limitations such as overwhelmed weak edges and the tendency to form loops.

Xu and Prince [10, 11] have proposed a new deformable model called Gradient Vector Flow (GVF) snake. Instead of using image gradients as the external force, it uses the spatial diffusion of the gradient of an edge map of the image. The amount of diffusion adapts according to the strength of edges to avoid distortion of object boundaries. This model solves both problems associated with initialization and poor convergence to boundary concavities. It revealed that external forces derived from the GVF snake has much larger capture range than the traditional snake.

The majority of the above mentioned snake models suffer from several limitations such as slow convergence speed and difficulty in adjusting weighting factors in energy terms. Menet [12] introduced the perception of B-spline snakes, emphasizing the advantages of local control, compact representation, and the possibility to include corners [13, 14]. Although this model addresses many of the problems associated with former models it is unable to fit into sharp corners of the object completely.

All of the snake models discussed above are incapable of increasing the local flexibility of the contour without adding more control points. A more flexible model called Dynamic NURBS (or D-NURBS), a physics based generalization of NURBS was introduced by Terzopoulos [15] for computer aided geometric design. This model has a capability to adjust the weight near a desired region interactively by the designer. Several good modifications and extensions to this model have appeared in recent parametric active contour literature.

Many aspects of the original model have been modified and extended by many researchers as discussed above. The latest snake models have a larger capture range and stronger convergence ability towards boundary concavities than a traditional snake. But unfortunately it has been observed that none of the above implementations provide a satisfactory solution to complex objects with sharp corners where the active contour tends to surpass the sharp corners of the object during deformation.

### III. PROBLEMS WITH SHARP CORNERS

#### A. Inadequate Number of Control Points

The traditional snake model, requires the operator to manually define the initial contour with several control points located quite closer to the object boundary. Thus, the initial contour depends on the user defined points that are placed around the desired object. If the initial contour is drawn with a fewer number of control points, or if two adjacent control points are situated at a large distance apart near sharp corners of the object, there is a high probability of the snake surpassing the sharp corners of the object during deformation.

Fig.1 illustrates the final position of the snake by running the original Kass snake with a real image of a tree leaf, both with the same number of iterations and similar values for the parameters  $\alpha$ ,  $\beta$ , and  $\gamma$ . In both cases all the control points were placed quite closer to the desired object manually by the operator. In Fig. 1(a), the initial contour was plotted using 15 control points which are somewhat equally spaced whereas Fig. 1(b), the initial contour was plotted using 25 control points with sufficient numbers of control points placed near sharp peak of the tree leaf at the left. The number of control points is maintained constant in both cases during evolution.

Fig. 1(a) clearly indicates that snake drawn using 15 control points, is not capable of capturing sharp corners and as a result, the active contour penetrated into the object. Though the initial contour is plotted closer to the object, it cannot cover the sharp corners due to a fewer number of points near the corner. On the other hand, as shown in Fig. 1(b), the snake was able to extract the nearest boundary of the desired object including the sharp corner at the left due to increased number of control points near this region.

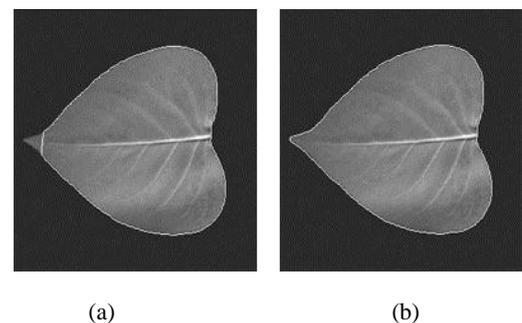


Fig. 1. Convergence of the active contour based on initial number of control points. (a): Initial contour drawn with 15 control points and (b): Initial contour drawn with 25 control points.

#### B. Strength of the Elasticity Force

The issue of surpassing corner points arises due to the strength of the elasticity force [16] as well. The moving equation for the parametric deformable models has been derived through the energy function which is a composition of elasticity force, bending force, image force and the constraint force. The elasticity force tries to minimize the distance between the control points by keeping them equidistant along the contour and it has the effect of causing the contour to shrink. If the distance between two adjacent control points is too high, especially

near a corner, the elasticity force tries to drag the control point away from the corner as shown in Fig. 2.

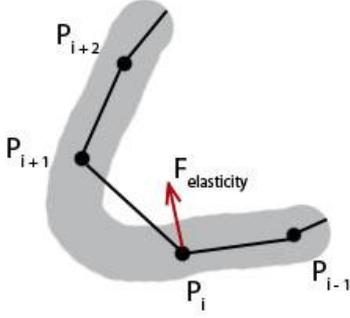


Fig. 2. The elasticity force  $F_{elasticity}$  drags the control point into the object.

Therefore, the accurate segmentation results are highly dependent on initial control points used to define the initial contour. Images with sharp corners may give rise to spurious convergence that surpasses the corner before it reaches the desired solution.

#### IV. METHODOLOGY

The main principle of energy minimizing formulation of deformable models is to find a parameterized curve that minimizes the weighted sum of internal energy, image energy, and constraint energy. The internal energy is defined within the curve itself and the image energy is computed from the image data.

If a snake is defined as a parametric curve  $\mathbf{v}(s) = (x(s), y(s))^T$ , where  $x$  and  $y$  are the coordinate functions, the total energy of the snake,  $E_{snake}$ , is defined as:

$$E_{snake} = \int_0^1 [E_{internal}(\mathbf{v}(s)) + E_{image}(\mathbf{v}(s)) + E_{con}(\mathbf{v}(s))] ds. \quad (1)$$

where  $E_{internal}$  represents the internal energy of the snake due to bending,  $E_{image}$  refers to image energy and  $E_{con}$  is the external constraint energy. The sum of the image energy  $E_{image}$  and the external constraint energy  $E_{con}$  is also known as the external energy. The internal energy of the snake is written as;

$$E_{internal} = (\alpha(s) |v_s(s)|^2 + \beta(s) |v_{ss}(s)|^2) / 2 \quad (2)$$

where the first order term  $|v_s(s)|^2$  gives a measure of the elasticity and the second order term  $|v_{ss}(s)|^2$  gives a measure of the curvature of the deforming snake. The influence that these terms have on the overall snake is governed by  $\alpha(s)$  that controls the tension of the contour while  $\beta(s)$  controls the rigidity. Therefore, the internal force holds the curve together (first order term) and keeps it from bending too much (second order term).

The image energy  $E_{image}$  (also referred to as potential energy) is derived from the image intensity data. It attracts the snake towards desired features in the image and settles on local minima at the image intensity edges occurring at object boundary. The image energy of the snake is written as;

$$E_{image} = -|\nabla I(x, y)|^2 \quad (3)$$

where  $\nabla$  denotes the gradient operator. In order to remove noise from the image and to increase the capture range of the snake, the image can be convolved with a Gaussian kernel before computing the gradient.

A new technique is proposed in this paper to incorporate prior knowledge about the object corners into the initial contour using the Harris operator to detect corners and the original Kass algorithm is used to implement the behavior of the active contour. The following three steps are taken at the beginning of the deformation process of the snake.

##### Step 1: Detection of Corners using Harris Operator

The proposed algorithm starts by detecting corner points on the desired object using Harris corner detector. The Harris operator is based on the local auto-correlation function of a signal where the function measures the local change of the signal with patches  $N_0$  shifted by a small amount in different directions. Given a shift  $(\Delta x, \Delta y)$  and a point  $(x, y)$ , the auto-correlation function is defined as;

$$C(x, y) = \sum_w [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \quad (4)$$

Where  $I(\cdot, \cdot)$  denotes the image function and  $(x_i, y_i)$  are the points in the window  $w$  (Gaussian) centered on  $(x, y)$ .

Let  $A$  be a  $2 \times 2$  matrix computed from image derivatives and is defined as

$$A = \sum_{(x,y) \in N_0} \nabla I(x, y) \nabla I(x, y)^T \quad (5)$$

(1) Let  $\lambda_1$  and  $\lambda_2$  be the eigenvalue of  $A$ . The Harris corner detector  $H$  can then be defined as;

$$H(x, y) = \lambda_1 \lambda_2 / (\lambda_1 + \lambda_2) \quad (6)$$

During the first step, the desired object should be selected. This selection of the area is based on the prior knowledge of the desired object or an interactive manual placement by the operator. Then, all the significant corners of the desired object are detected using Harris operator and the points are placed in an array called `harris_points [ ]` which is defined as:

$$\text{harris\_points [ ]} = \{ \mathbf{P}_i(x, y) | i = 0, 1, 2, \dots, n - 1 \} \quad (7)$$

##### Step 2: Defining Additional Control Points

The required additional control points are placed manually by the operator somewhat closer to the desired object and these additional control points are placed in a separate array defined as follows:

$$\text{additional\_points [ ]} = \{ \mathbf{P}_i(x, y) | i = 0, 1, 2, \dots, m - 1 \} \quad (8)$$

### Step 3: Reconstruction of the Initial Contour

At the next step, both arrays are merged together and a new array is constructed as follows:

$$\text{all\_points [ ]} = \{P_i(x,y) | i = 0,1,2,\dots, n+m-1\} \quad (9)$$

This new array contains both the corner points detected by the Harris operator and the additional control points given by the operator in an unordered manner.

Before the contour deforms, in order to reconstruct the initial contour, it rearranges all points according to the correct order by considering the clockwise angle as in the following algorithm

#### ALGORITHM: Reconstruction of the initial contour

---

```

// n is the total number of corner points detected by the
Harris detector
// m is the total number of additional points defined by the
operator

mid_point = mid point of the desired object
reference_point = a point corresponds to the mid point
θ = angle corresponds to the reference point, middle point
and current point
Load (image)
// Detection of corners using Harris operator
harris_points [ ] = getCornertPoints (execute Harris
operator)

// Detection of additional control points
additional_points [ ] = getControlPoints (mouse input)

// merge two arrays
all_points [ ] = harris_points [ ] + additional_points [ ]

// Reconstruction of the initial contour
for (i = 0 to (n+m)-1) {
    current_point = ith point
    A2 = distance2(middle_point , reference_point)
    B2 = distance2(middle_point , current_point)
    C2 = distance2(reference_point , current_point)
    θ = arccos (( A2 + B2 - C2) / 2AC)
    if (θ > 180°)
        θ = 360° - θ
    end if
    theta_array [ ] = {Pi (θ angle) , (x , y)}
}
end for

//sort the array by the θ angle according to the ascending
order
final_points [ ] = Sort (theta_array [ ] , ascending)
//reconstructed initial contour will be the inputs to the
snake algorithm
Contour = Snake Algorithm (final_points [ ])
    
```

---

## V. EXPERIMENTAL RESULTS

This section presents the results obtained from implementation of the proposed algorithm on the Matlab version R2012a without code optimization, and executed

on an Intel Core 2 Duo 2.66 GHz standard desktop computer. For the experimental purpose, the new technique was applied to original Kass algorithm.

### A. Experiments with synthetic images

The convergence of the algorithm was tested on many synthetic images where the region of interest consists of several sharp corners. The final extracted boundary using the proposed algorithm on the synthetic image is depicted in Fig. 3. where the proposed algorithm gave the nearest actual boundary of the object (while capturing all the sharp corners accurately) compared with the former methods. The model parameters were empirically selected as  $\alpha = 0.40$ ,  $\beta = 0.20$ , and  $\gamma = 1.00$ . For all the images, the Gaussian filter is applied on the image to boost the captured range.

It was also noticed that the proposed snake behaves in a quite different manner while it deforms. The detected corner points tend to stay in the same location without any movement since the image energy is a minimum at those places. Therefore, particular advantage of this technique over the traditional snake is its ability to capture sharp corners precisely.

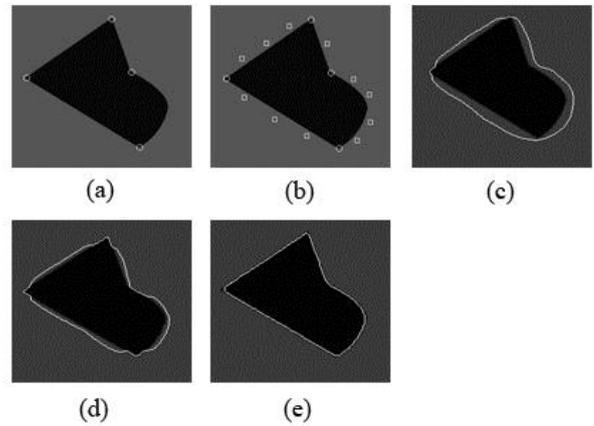


Fig. 3. Initialization and deformation of the snake on a synthetic image: (a) corner points detected using Harris operator, (b) additional control points defined by the operator, (c) initial contour after reconstruction, (d) the location of the contour after 100 iterations and (e) after 200 iterations.

We have further analyzed the convergence process by using standard distance error as proposed in [17] as this method has also been widely used to quantify segmentation results obtained from active contours. Several synthetic images including objects with sharp corners were selected as seen in Fig. 4. The initial contour was first plotted using 20 control points for all the cases and the Gaussian filter is applied on the images to boost the captured range.

As seen in Fig. 4, the final output shows that the boundary detection was achieved best with the proposed algorithm even when the snake was initialized with a fewer number of control points for an object with sharp corners.

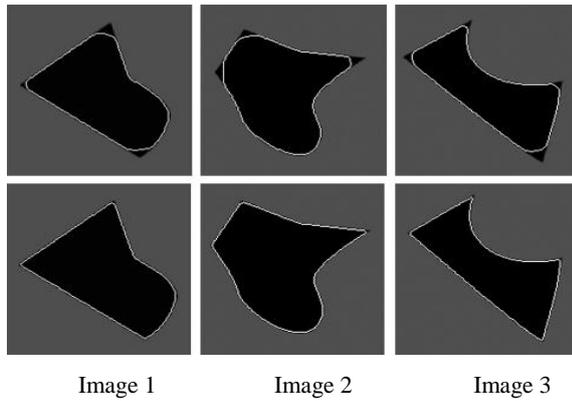


Fig. 4. Convergence of the active contour on synthetic images. (top row): Kass algorithm and (bottom row): proposed algorithm.

TABLE I. PERFORMANCE CAMPARIOSN BETWEEN THE ORIGINAL

Image	Standard Distance Error with Kass Algorithm	Standard Distance Error with Proposed Algorithm
Image 1	1.9674	1.3766
Image 2	1.7807	1.1262
Image 3	2.1067	1.1663

Standard distance errors with respect to the actual boundary of the object were calculated to find the best approximates for the perimeter of the object as depicted in Table I where it is evident that the proposed algorithm performs better than the original Kass algorithm. In other words, experimental results clearly indicate the increased capability of the proposed algorithm in extracting the nearest boundary of the desired object having sharp corners.

*B. Experiments with real images*

The proposed algorithm was further tested on real images. All the strong attraction centers (also known as false feature points) inside the desired object also detected as seen in Fig. 5 (fossil of a shark tooth, <http://www.blackhawkfossils.com>). Due to such false feature points, the snake moves into the object by surpassing the actual boundary while it deforms. Therefore, the proposed algorithm tends to end up with false segmentation results for real images. In order to segment real images, the proposed algorithm should be improved further by avoiding all the false feature points situated inside the object.

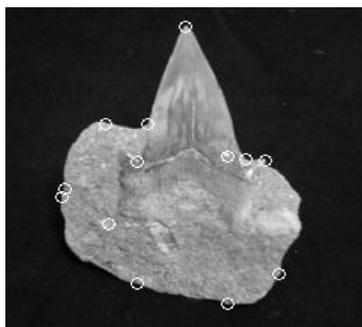


Fig. 5. Some key points in the inner part of the object also detected

VI. CONCLUSION

Though active contours are useful in many applications, most of the models are not capable of capturing precisely the boundary of complex objects with sharp corners. In this paper, a new technique is proposed to reconstruct the initial contour by incorporating prior knowledge of the significant corners of the object of interest detected using the Harris operator. The results obtained after applying the technique on several synthetic images show the increased performance of the proposed method compared with traditional active contour models. Further studies on improving the algorithm to achieve accurate results on real images need to be explored.

REFERENCES

- [1] T.McInerney and D.Terzopoulos. “Deformable models in medical image analysis”: A survey. *Medical Image Analysis*, vol. 1, pp. 91-108, 1996.
- [2] A. Singh, D. Goldgof, and D. Terzopoulos, editors. “Deformable models in Medical Image Analysis”. IEEE Computer Society Press, 1998.
- [3] R. Hegadi, A. Kop, and M. Hangarge. “A Survey on Deformable Model and its applications to medical imaging”, *IJCA Special issue on Recent Trends in Image Proc. and Pattern Recog. (RTIPPR)*, pp. 64-74, 2010.
- [4] P. Andrey, T. Boudier, “Adaptive active contours (snakes) for the segmentation of complex structures in biological images”, in *Proc. ImageJ Conference*, pp. 18-19, 2006.
- [5] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models”, *Int. Journal of Computer Vision*, vol. 1, pp. 321-331, 1988.
- [6] J. S. Duncan, N. Ayache, “Medical Image Analysis: Progress over Two Decades and the Challenges Ahead”, in *Proc. IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 90-92, 2000.
- [7] C. Harris and M. Stephens, “A combined corner and edge detector”, in *Proc. 4th Alvey Vision Conference*, pp. 147-151, 1988.
- [8] T. McInerney, D. Terzopoulos, “T-sankes: Topology adaptive snakes”, in *Proc. International Conference on Computer Vision*, pp. 840-845, 1995.
- [9] L. D. Cohen, “On active contour models and balloons”, *CVGIP: Image Understanding*, 53(2), pp. 221-218, 1991.
- [10] C. Xu and J. L. Prince, “Gradient Vector Flow: A New External Force for Snakes”, in *Proc. IEEE Conf. on Computer Vision and pattern Recognition (CVPR)*, Los Alamitos: Comp. Soc. Press, pp. 66-71, 1997.
- [11] C. Xu and J. L. Prince, “Snakes, Shapes, and Gradient Vector Flow”, *IEEE Transaction on Image Processing*, vol.7, no. 3, pp. 359-369, 1998.
- [12] S. Menet, P. Saint-Marc, and G. Medioni. “B-snakes: Implementation and application to stereo”. In *proceedings DARPA*, pp. 720-726, 1990.
- [13] P. Brigger and M. Unser, “Multi-scale B-spline Snakes for General Contour Detection”, in *Proc. Wavelet Applications in Signal and Image Processing VI, SPIE vol. 3458*, pp. 92-102, 1998.
- [14] P. Brigger, J. Hoeg and M. Unser, “B-Spline Snakes: A Flexible Tool for Parametric Contour Detection”, *IEEE Trans. on Image Processing*, vol. 9, pp. 1484-1496, 2000
- [15] H. Qin, D. Terzopoulos, “D-NURBS: a physics-based framework for geometric design”, *IEEE Transactions on Visualization and Computer Graphics* vol. 2 (1) (1996) 85-95.
- [16] J. Ahlberg, “An Active Contour in Three Dimensions”, Thesis project at Computer Vision Laboratory, Linkoping University, pp. 17, 1996.
- [17] L. Staib, J. Duncan, “Model-based deformable surface for medical images”, *IEEE Transactions on Medical Imaging*, 1, pp. 859-870, 1996.