Dynamic Node Distribution in Adaptive Snakes for Road Extraction

Peggy Agouris Sotirios Gyftakis Anthony Stefanidis
Dept. of Spatial Information Engineering and
National Center for Geographic Information and Analysis
University of Maine
5711 Boardman Hall #348
Orono, ME 04469-5711, USA
{peggy, sotiris, tony}@spatial.maine.edu

Abstract

In this paper we address the issue of road extraction from digital imagery using deformable contour models (snakes). We present a novel variation of the traditional snakes solution, where additional nodes are inserted, and redundant nodes are deleted, to better describe the complexity of the extracted line. Node insertion and deletion are based on an analysis of the energy terms of the snake solution. This allows us to use more, closely spaced nodes along the high curvature areas of a road, compared to the linear segments of the same road outline. This dynamic manipulation of the number and spacing of nodes within a single snake allows us to better capture the geometry of the road, and to better accommodate its radiometric behavior. Thus it produces more accurate results than the traditional snake solution. Here we present our approach and experimental results to demonstrate its performance for road extraction in geospatial applications.

1. Introduction

Object extraction from digital imagery is a fundamental computer vision operation. In geospatial applications its role is to populate and update GIS databases with precise information on the positions of man-made objects (e.g. roads, buildings) and natural features (e.g. rivers) [1, 7, 8, 12]. This commonly involves the analysis of digital imagery captured by aerial or satellite sensors.

A variety of approaches have been proposed for object extraction, with varying degrees of success. It is common knowledge that there does not exist a single approach outperforming all alternatives. Instead, it is often the nature of the application at hand that renders certain approaches more suitable for the specific task.

Deformable contour models (snakes) are among the most popular object extraction approaches. The framework of this approach was originally introduced by Kass *et al* in [9]. The concept of snakes is based on curve detection through an optimization process. This optimization makes use of models of curve contrast and smoothness that employ elastodynamic models and descriptions of their behavior under the application of external and internal forces. Other researchers have also connected images with energy. For example, as stated in [6] any image can be considered as an isolated thermodynamical system by identifying the image intensity with some thermodynamical variable. Since its introduction, the deformable model has been implemented in a variety of techniques. The various approaches of modeling a snake are based on Fourier transform, B-splines, finite elements, discrete representations, dynamic programming, or least squares [10]. An attempt to unify the most important variations of snakes within the framework of a general finite element formulation is presented in [11].

Main challenges in the application of the snake method are related to the initialization of the contour and the correct convergence of the solution. Various methods have been proposed addressing these problems. They include the use of multi-scale approaches [13], pressure forces [4], finite-element methods [5], level set [14], multi-level [2], and gradient vector flow [17].

Our work in this paper is focusing on road extraction from small scale aerial imagery using snakes. Snakes are highly suitable for such application, as they allow us to overcome certain inherent problems like occlusions, width changes, surface material variations, and the effects of overpasses and intersections.

In this paper we present a novel variation of the traditional snakes solution, where additional nodes are inserted, and redundant nodes are deleted, to better describe the complexity of the extracted line. This dynamic distribution of nodes is based on an analysis of the energy terms of the snake solution. This allows us to use more, closely spaced nodes along the high curvature areas of a road, compared to the linear segments of the same road outline. The dynamic manipulation of the number and spacing of nodes within a single snake allows us to better capture the geometry of the road, and to better accommodate its radiometric behavior. Thus it produces more accurate results than the traditional snake solution. Here we present our approach and experimental results to demonstrate its performance for road extraction in geospatial applications

The paper is structured as follows. In Section 2 we present an overview of the traditional snakes solution, focusing on the relevant energy terms. In Section 3 we present our approach for the dynamic positioning of snake nodes using by analyzing the snake energy terms. Experimental results are presented in Section 4 to demonstrate the performance of our approach for road extraction in geospatial applications. We conclude with comments in Section 5.

2 Snake model

2.1 Energy functions

In general, the energy function of a snake contains descriptions of internal and external forces, as well as external constraints. The internal forces allow the contour to stretch or bend at the specific point, while maintaining certain smoothness and continuity. The external force attracts the contour to significant features on the image (namely the road location in it), while the external constraints represent user-imposed restrictions.

The total energy of each point is expressed as a sum of individual energy terms:

$$E_{snake} = \alpha \cdot E_{cont} + \beta \cdot E_{curv} + \gamma \cdot E_{edge} \qquad (1)$$

where E_{cont}, and E_{curv} are the first and second order continuity constraints (internal snake forces), E_{edge} is the edge strength (external snake force), and α, β, γ are relative weights of each energy term. Internal forces tend to produce smooth snake curves, while the external force attracts the snake to edge locations in the image.

2.2 Continuity term

If $v_i(x_i,y_i)$ is a point on the contour, the first energy term in eq.(1) is defined as follows:

$$E_{cont} = d_{av} - |v_i - v_{i-1}| \qquad (2)$$

with d_{av} the average distance between points, defined for a snake with n points as:

$$d_{av} = \sum_{i=1,n-1} |v_{i+1} - v_i| / (n-1) \qquad (3)$$

The continuity energy term guarantees that snake points will be evenly spaced, while minimizing their distance [16].

2.3 Curvature term

This is an estimation of the snake's second derivative and is calculated as:

$$E_{curv} = |v_{i-1} - 2v_{i-1} + v_{i+1}|^2 \qquad (4)$$

Since the continuity term produces evenly spaced points, the above term gives the snake curvature multiplied by a constant. This constant becomes insignificant, because the curvature term is normalized in the neighborhood of each point.

2.4 Edge term

This energy term describes the external force that attracts the snake to the road location in the image. In general, it forces points to move towards image edges. An expression of this term may be provided by:

$$E_{edge} = -\nabla I(v_i) \qquad (5)$$

where $I(v_i)$ is the image function (gray values) at snake point v_i. We use the negative sign to attract the snake to image points with high gradient values. Since the gradient is a metric for image edges, the snake is attracted to strong edge points. The image gradient at each point is normalized to handle even small gray value variations at the neighborhood of that point.

2.5 Optimization procedure

After the initialization of the snake contour, the points along the contour move to new energy minimum locations. These new locations are found through an iterative procedure. In each iteration, an optimization process is used to compute the new snake location.

A simple and stable solution to the optimization proble was introduced by Amini *et al* in [3] using dynamic programming. An alternative, approach is suggested by Williams and Shah in [16]. This method (greedy algorithm) is found to be faster than dynamic programming and more stable and flexible than the variational calculus approach. This is the method used in our implementation of snakes.

An aspect that influences the optimization procedure is the resolution of the snake. This resolution characterizes the speed of the optimization. The smaller the resolution, the slower the optimization performs. In the initialization of the snake the user specifies the initial k nodes. Given a value d1 (in pixels), n points are added between the two existing nodes, if their distance D1 is greater than d1. Hence, n=integer part(D1/d1) and d1 is the resolution of the snake.

3. Proposed method

3.1 General

In our approach we assume the use of small scale image aerial imagery from which we have extracted a road segment using snakes with a low interpolation resolution. Our main objective is to improve the accuracy of road extraction by selectively intensifying the interpolation resolution. By having nodes equally distributed, common snakes solutions do not take into account the geometric complexity of the extracted object. This lowers the accuracy of the extracted representation. In road extraction this is commonly manifested by extracting a polygonic approximation that deviates from the actual road by few pixels. To overcome such problems we propose an adaptive iterative snake solution where the snake solution of an iteration is analyzed to:

- remove redundant nodes, and to
- densify locally the solution as necessary, to better capture the geometric complexity of the extracted feature.

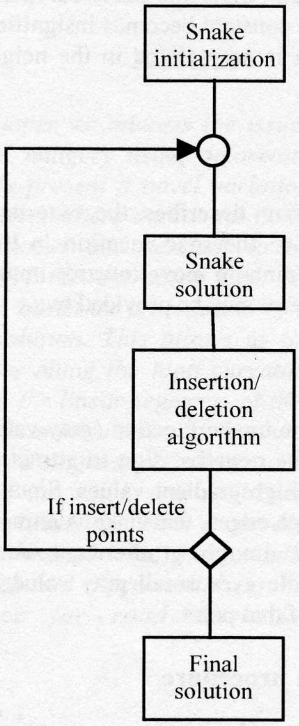

Figure 1. General flowchart of our approach

Our argument is that the quality of the snake solution is inherently described by the local values of its corresponding energy terms. Accordingly, in our iterative process we analyze this information at the end of each snake iteration to define the distribution of nodes in the next round.

More specifically, the output result of a snake solution includes the optimized coordinates of snake nodes, the energy values of each point and its components (continuity, curvature and edge). These energy components indicate how well the snake has performed on each point.

The general flowchart of the proposed method is shown in figure 1. Our algorithm will remove points that are redundant for road description and will insert points is areas that the description of the road is not adequate. The basic procedures that will perform these tasks are *deletion* and *insertion,* followed by a variation of snake in which only few points (the manipulated ones) will be permitted to move. In our approach we make use of two general energy thresholds: low energy threshold (EnLow) and high energy threshold (EnHigh). These thresholds are used to identify candidates for further processing as outliers in our solution.

3.2 Node Deletion

The flowchart of this operation is shown in figure 2. The purpose of node *deletion* is to improve the efficiency of road extraction by removing redundant points without lowering the extraction accuracy. In order to achieve this we set the following rules for node deletion (assuming that radiometry describes the gradient values):

1) the radiometric information is good , *and*
2) the curvature term is very low.

The first condition states that there is a great probability that this point belongs on the road edge. Good radiometry is implied from (absolute) high edge term (equation 5). If E_{edge} is the radiometry term, we define the threshold for good radiometry (E_{edge}ThrHigh) as:

$$E_{edge}ThrHigh = \min(E_{edge}) + range(E_{edge})*EnHigh \tag{6}$$

Points that have E_{edge} greater than E_{edge}ThrHigh are considered to have good radiometry and are candidates for deletion.

The second rule guarantees that the geometry of the snake will be distracted minimally. The smoothness of the snake is described in the curvature term (equation 4). If the curvature of the point is low, it implies that the snake is very smooth (close to straight line) in that area. If E_{curv} is the curvature term, we define the threshold for low curvature (E_{curv}ThrLow) as:

$$E_{curv}ThrLow = \min(E_{curv}) + range(E_{curv})*EnLow \tag{7}$$

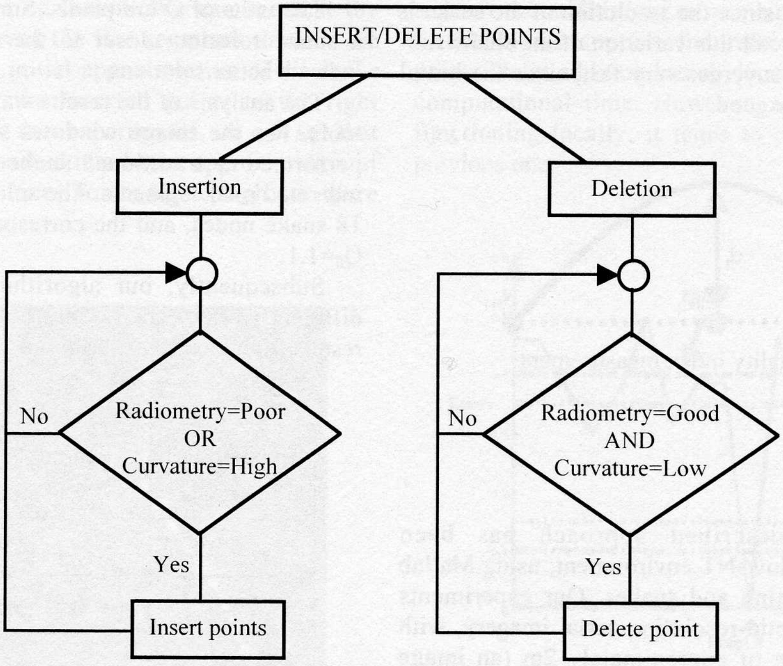

Figure 2. Deletion/Insertion flowchart

If any points have E_{curv} lower than E_{curv}ThrLow, they are candidates for deletion, as their curvature is very low.

Finally, we combine the above two rules and we select the candidate points that satisfy these rules, for deletion. Note that we demand both of the rules to be true for points deletion. The next step is to insert points between the remaining snake points.

3.3 Node insertion

The flowchart of this operation is presented in figure 2. The objective of this operation is to improve the accuracy of object extraction by inserting additional nodes in areas where the snake has not performed well or in areas where the snake resolution was not sufficient to follow the actual road curvature. In those areas we increase the snake resolution by adding more points. The rules for points insertion are the following:

1) the radiometry is poor, *or*
2) the curvature is high.

The first condition is necessary for areas where there are gaps in the continuity of the road. As in the previous subsection, poor radiometry is implied from low edge term (equation 5). The threshold for poor radiometry is the same(E_{edge}ThrHigh) as in equation 6. This time we consider points that have E_{edge} lower than E_{edge}ThrHigh as candidates for points insertion.

The second condition is needed in the areas where the snake resolution is not large enough to follow the curvature of the road. In those areas the curvature of the snake is high. Similarly to equation (7) we define the threshold for high curvature (E_{curv}ThrHigh) as:

$$E_{curv}\text{ThrHigh} = \min(E_{curv}) + \text{range}(E_{curv})*\text{EnHigh} \qquad (8)$$

If any points have E_{curv} higher than E_{curv}ThrHigh, they are candidates for insertion, as their curvature is very high.

In case that anyone of those criteria is satisfied, then two additional points are inserted; one on each side of the point under investigation by interpolating the specific point coordinates with each one of its neighbors.

As indicated in equations (6)-(8), our apporach makes use of the energy thresholds EnHigh and EnLow. These thresholds can be defined by a user, or they may be selected automatically through a statistical analysis of the snake solution data. In the experiments section of this paper we investigate how do these thresholds affect the snake accuracy.

3.4 Sub-snake

After the procedures of deletion and insertion, we employ a variation of the snake solution using the remaining points. In this variation we allow only selected points to move, namely added nodes and relevant neighboring nodes. We assume that there is no need to change the positions of the rest of the snake nodes. Comparing to the traditional snake method, the change is that we do not use the average distance component (equation (3)) anymore, since the resolution of the snake is no longer uniform. We call this variation of the snake sub-snake. The sub-snake converges very fast, since the initial approximation is already good.

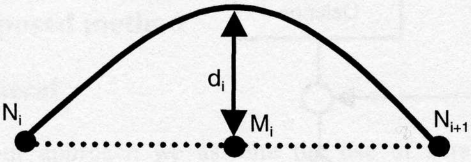

Figure 3. Quality index measurement

4. Experiments

The previously described approach has been implemented in a WindowsNT environment, using Matlab for dynamic programming and snakes. Our experiments were performed using mid-resolution aerial imagery, with ground pixel resolution of approximately 2m (an image pixel corresponds to a 2m x 2m patch on the ground). The figures used in this section show 500x500 pixel windows from such imagery. The images shown in the next figures were downloaded from the BADGER System home page (http://badger.parl.com/), which is a software system for delivering image and map data of the San Francisco Bay Area via the Internet.

In order to evaluate the performance of the proposed approach, we need to define a quality index for the accuracy of the extraction. Criteria for accuracy measurement have been proposed in [15]. Similarly, we define an index that evaluates the geometric accuracy and is based on the average geometric error of a snake solution.

Experiment #	EnLow	EnHigh
1	20	80
2	30	70

Table 1. Experiment parameters

For each line segment (N_i-N_{i+1}) between neighboring snake points (figure 3), we identify the midpoint (M_i). Next, we calculate the distance (d_i) of this midpoint (M_i) from the road and we define the quality index (Q) as the mean of all these distances:

$$Q = \Sigma_{i=1,n-1} \; d_i/(n-1) \qquad (9)$$

The units of Q are pixels. Smaller values of Q indicate a snake solution closer to the true road, and therefore indicate better solutions.

The analysis of the results was performed using Matlab tools. For the image windows shown in Figures 4-6 we performed a traditional snake solution to extract the indicated road segments. The solution (Fig. 4) made use of 18 snake nodes, and the corresponding quality index was Q_0=1.1.

Subsequently, our algorithm was tested using two different pairs of energy thresholds (Table 1). Experiment results from these two variations are presented in Table 2.

Experiment #	Total points	Q	Change of Q
1	23	0.9	18%
2	38	0.6	45%

Table 2. Experiment results

The resulting solutions for experiments 1 and 2 are shown in figures 5 and 6 respectively. As we can see, the two experiments have resulted in accuracy improvements of 18% and 45%, with an increase in the number of nodes by 27% and 110% respectively.

Figure 4. Snake solution before deletion insertion

In experiment 2 we increase the lower energy threshold and we decrease the high energy threshold. This change results in more deleted points and more inserted ones. As we expected the inserted points in difficult areas have improved the accuracy of the snake (Q_2<Q_1). In figure 6, a zoom in a difficult area is shown. We can see that the inserted points have moved the line segments towards the correct road edges. The initial approximation of the snake was adequate to drive the inserted points to the right direction. Comparing the two experiments, we can see that the second set-up has resulted in a better approximation of the road by introducing more points in the high curvature region of the road.

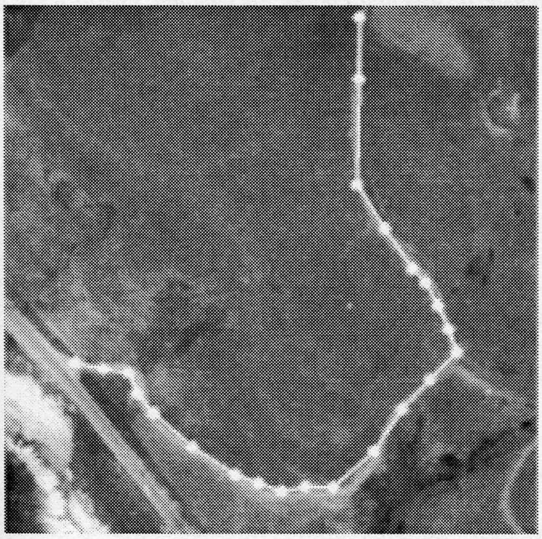

Figure 5. Solution after deletion/insertion (20/80)

Figure 6. Solution after deletion/insertion (30/70)

In figure 7 we show the curvature energy of the initial extraction plus the thresholds of the two experiments. By lowering the high energy threshold in exp. #2 we tend to support the addition of more points. By raising the lower threshold we support the removal of more redundant points. Point addition has obvious connotations in terms of computational time. However, as each new iteration is functioning locally, it tends to converge faster than the previous one.

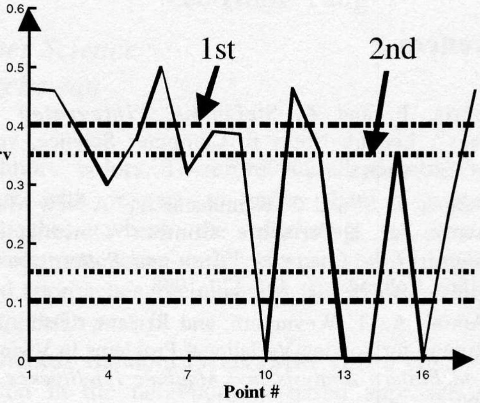

Figure 7. Curvature energy and thresholds

Similar experiments have been performed with numerous images and road segments. The results used to demonstrate the performance of our approach in this paper are representative of an average solution. Indeed, in our experiments the average improvement for the configuration of experiment #1 was 16%, and 39% for configuration #2. The average number of iterations following a snake solution is between 2 and 3.

5. Conclusions

In this paper we presented a new method for improving road extraction accuracy using snakes. Our approach is based on an iterative application of snakes, where additional nodes are inserted, and redundant nodes are deleted, to better describe the complexity of the extracted line. This dynamic manipulation of node distribution is based on an analysis of the energy terms of the snake solution.

Experimental results confirmed an improvement of the geometric accuracy of the extracted outline, in the order of 15% - 40%. This variation depends on the selected energy thresholds. By narrowing the energy range of the accepted points we increase the snake resolution, and hence the accuracy of the sub-snake. The obvious trade-off is that by narrowing the acceptable energy range we increase the number of iterations and correspondingly the solution runtime.

This novel variation of traditional snakes produces more accurate results than the traditional solution. We are currently working on the incorporation of our approach in commercially-available photogrammetric software.

Acknowledgments

This work was supported by the National Science Foundation through CAREER grant number IIS-9702233, and by BAE Systems.

References

[1] Agouris, P. and A. Stefanidis, *"Integrated Spatial Databases"*, Lecture Notes in Computer Science, vol. 1737, Springer Verlag, Berlin, 2000.

[2] Akgul, Y. S. and C. Kambhamettu, "A New Multi-Level Framework for Deformable Contour Optimization," in *Proceedings of the Computer Vision and Pattern Recognition*, Fort Collins, Colorado, pp. 465-470, 1998.

[3] Amini, A., T. Weymouth, and R. Jain, "Using Dynamic Programming for Solving Variational Problems in Vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12:855-867, 1990.

[4] Cohen, L. D., "On Active Contour Models and Ballons," *CVGIP: Image Understanding*, 53:211-218, 1991.

[5] Cohen, L. D. and I. Cohen, "Finite element method for active contour models and balloons for 2D and 3D images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:1131-1147, 1993.

[6] Ferraro, M., G. Boccignone, and T. Caelli, "On the Representation of Image Structures via Scale Space Entropy Conditions," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21:1199-1203, 1999.

[7] Gruen, A., E. Baltsavias, and O. Henricsson, *"Automatic Extraction of Man-Made Objects from Aerial and Space Images II"*. Birkhaeuser, Basel, 1997.

[8] Gruen, A., O. Kuebler, and P. Agouris, *"Automatic Extraction of Man-Made Objects from Aerial and Space Images"*. Birkhaeuser, Basel, 1995.

[9] Kass, M., A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," in *Proceedings of the 1st Int. Conf. on Computer Vision*, London, pp. 259-268, 1987.

[10] Li, H., "*Semi-Automatic Road Extraction from Satellite and Aerial Images*", Ph.D. Thesis, Zürich, Switzerland, ETH, 1997.

[11] Liang, J., T. McInerney, and D. Terzopoulos, "United Snakes," in *Proceedings of the IEEE International Conference on Computer Vision*, Corfu, Greece, pp. 923-940, 1999.

[12] Lukes, G., *"Proceedings Image Understanding Workshop"*, Morgan Kaufmann, San Francisco, 1998.

[13] Mahmoodi, S., B. S. Sharif, and E. G. Chester, "Contour Detection Using Multi-Scale Active Shape Models," in *Proceedings of the International Conference on Image Processing*, Washington, DC, 1997.

[14] Wang, X. and W. G. Wee, "A New Deformable Contour Method," in *Proceedings of the 10th International Conference on Image Analysis and Processing*, Venice, Italy, pp. 430-435, 1999.

[15] Wiedemann, C., C. Heipke, H. Mayer, and O. Jamet, "Empirical Evaluation of Automatically Extracted Road Axes," in *Empirical Evaluation Methods in Computer Vision*, K. J. Bowyer and P. J. Phillips, Eds.: IEEE Computer Society Press, pp. 172-187, 1998.

[16] Williams, D. J. and M. Shah, "A Fast Algorithm for Active Contours and Curvature Estimation," *CVGIP: Image Understanding*, 55:14-26, 1992.

[17] Xu, C. and J. L. Prince, "Gradient Vector Flow: A New External Force for Snakes," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, Puerto Rico, pp. 66-71, 1997.