Snake-based Technique for Automated Coronal Loop Segmentation

Jong Kwan Lee
Department of Computer Science
Bowling Green State University
leej@bgsu.edu

Woon Khang Tang
Department of Computer Science
Bowling Green State University
woonkhang@gmail.com

ABSTRACT

A new approach to automatically segment the solar coronal loop structures from intensity images of the Sun's corona is described. The approach is based on the active contour models (snakes) and exploits the Gaussian-like shape of the coronal loop cross-sectional intensity profile to refine the snake's position. The approach utilizes the principal component analysis to automatically initialize the snake's position. It then uses a greedy minimization method to attract the snake toward the center of coronal loop structures in each image. Its effectiveness is evaluated through experiments on synthetic and real images.

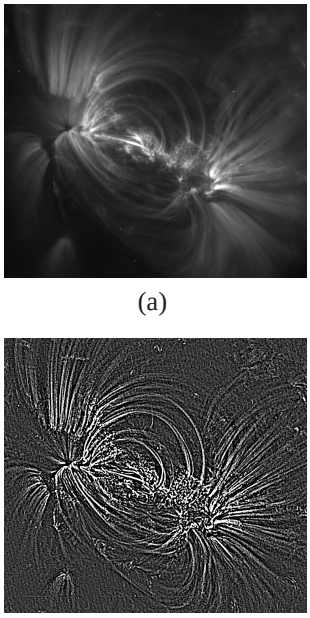
Keywords: Active contours, Solar coronal loops segmentation, Image processing

1 INTRODUCTION

Automated feature segmentation techniques have been used and studied widely in many applications (e.g., iris recognition [Teo05], fingerprint recognition [Pan01, Mag09], face recognition [Wis97, Kuk04], remote sensing [Cao09], etc.). Automated segmentation techniques can aid understanding of the feature characteristics. In this paper, we consider the problem of automatically segmenting the coronal loop structures from solar imagery.

The corona is the outermost layer of the Sun's atmosphere. Many solar scientists study the corona to gain insight of the solar activities (e.g., solar storms) that impact our geo-space environment. One common way to study the corona is via observation of the loop structures in the corona. These loop structures are the traces of the solar magnetic field that ultimately drives the Sun's dynamic activities [Lee06]. The typical way solar scientists observe the coronal loop structures is by considering them in solar imagery collected by a satellite. One preferred source of coronal images is NASA's TRACE satellite [Sch99]. (Details of TRACE image acquisition process are also explained in [Sch99] and omitted here.) A sample TRACE coronal image is shown in Figure 1 (a). In the figure, the bright arching structures are the coronal loops.

Automatic segmentation of the coronal loop structures is challenging as the structures have complex shapes and very blurry boundaries. Moreover, they tend to appear very close to each other or even overlap each other in the image. In addition, image noises and other non-loop features (e.g., sponge-like wide white spots) are present on the coronal images.



(a)

(b)

Figure 1: (a) TRACE coronal image and (b) Preprocessed image

In this paper, we present a new method, based on active contour models (i.e., also known as snakes) [Kas87], to fully-automatically segment the coronal loop structures from the TRACE images. The method uses the principal component analysis-based loop direction measures to initialize the snake position. It also exploits the shape of the coronal loop's

cross-sectional intensity profile to refine a greedy algorithm-based snake.

The paper is organized as follows. Section 2 discusses related work. In Section 3, the new snake-based approach for segmenting solar coronal loop structure is described. The experimental results of applying the new approach to synthetic and real coronal images are presented in Section 4. Section 5 concludes this paper.

2 RELATED WORK

In this section, the snake that our approach is based on and the prior efforts on automated coronal loop segmentation are discussed.

Kass et al. [Kas87] have introduced the snake as an energy minimizing contour that is represented parametrically as $v_i(s) = (x_i(s), y_i(s))$, where (x_i, y_i) is the i^{th} point position along the contour. An energy function is associated with the snake. Snake finds an image feature of interest (e.g., lines or edges) by minimizing its energy function. Snake's energy function (E_{snake}) is defined as the sum of internal energy term (E_{int}) and external energy term (E_{ext}) of each snake point and is defined as

$$E_{snake} = \int_0^1 E_{int}(v(s)) + E_{ext}(v(s))ds, \qquad (1)$$

where E_{ext} is the sum of image energy (E_{img}) and external constraint energy (E_{con}). E_{int} can be defined as

$$E_{int} = \frac{\alpha(s)|\mathbf{v_s}(s)|^2 + \beta(s)|\mathbf{v_{ss}}(s)|^2}{2}, \qquad (2)$$

where $\alpha(s)$ is the weight of the first-order term (i.e., $\mathbf{v_s}(s)$) and $\beta(s)$ is the weight of the second-order term (i.e., $\mathbf{v_{ss}}(s)$). $\mathbf{v_s}(s)$ and $\mathbf{v_{ss}}(s)$ terms prevent gaps and sharp corners. E_{img} can be expressed as a weighted combination of the line energy (E_{line}), the edge energy (E_{edge}), and the termination energy (E_{term}). E_{line} can force the snake to move toward high intensity points. E_{edge} can force the snake to move toward the image points with large image gradient. E_{term} can force the snake to move toward the end points of lines or edges. E_{con} can be defined as $E_{con} = -k(\mathbf{x_1} - \mathbf{x_2})^2$, where k is the spring constant, $\mathbf{x_1}$ is a snake point, and $\mathbf{x_2}$ is an image point.

Kass et al. [Kas87] have minimized the snake energy function iteratively using Euler equations expressed in a sparse matrix form. However, using their minimization method, some snake points tend to congregate in certain image points and the process can be unstable. Amini et al. [Ami90] have proposed a dynamic programming-based minimization process that can avoid some of Kass et al. [Kas87]'s problems. However, the dynamic programming-based snake approach can be slow. Williams and Shah [Wil92] have proposed a greedy algorithm for the snake's energy minimization which is more than an order of magnitude faster than the dynamic programming-based approach. In addition, it tends to be more stable. Here, we note that our new snake-based approach also uses the greedy algorithm-based approach.

Next, the prior works on automated coronal loop segmentation are discussed.

Oriented Connectivity Method (OCM) by Lee et al. [Lee06] is the first automated coronal loop segmentation method. OCM is a constructive edge linkage method that utilizes a simplified solar magnetic field estimate to guide the loop segmentation process. In particular, the magnetic field estimate is used to progressively link neighboring loop points together that have orientations that are consistent with the magnetic field's orientation.

Lee et al. [Lee06b] have also introduced another automated coronal loop segmentation method called the *Dynamic Aperture-based Method* (DAM). DAM segments coronal loops by exploiting the Gaussian-like shape of loop cross-sectional intensity profiles [Car03]. Specifically, DAM links the image points that have similar Gaussian shape parameters (as determined by a ruled Gaussian surface fitting on image points) and have similar loop orientation (as determined by the principal component analysis (PCA) [Dud00]).

Smith [Smi05] has adapted the *Unbiased Detection Method* (UDM) by Steger [Ste98] for coronal loop segmentation. UDM segments curvi-linear structures with different lateral contrast by taking account the geometry of the structure surroundings. It utilizes the second derivatives of image intensities in the direction perpendicular to the structure to find the centroid position of a curvi-linear structure.

Sellah and Nasraoui [Sel08] have utilized a type of randomized Hough Transform called *Incremental Random Hough Transform* (I-RHT) to detect coronal loop structures. (Randomized Hough Transform (RHT) [Kul90] is a fast Hough Transform approach that alleviates some of the expensive computational requirements of the standard Hough Transform (HT) [Hou62].) I-RHT detects a coronal loop structure by using a stream clustering algorithm to continuously update and extract the maximum bin of the HT accumulator in an incremental fashion. However, their approach was limited to detection of only one elliptical coronal loop structure.

Recently, Aschwanden [Asc10] has introduced a coronal loop tracing technique called *Oriented Coronal CUrved Loop Tracing* (OCCULT), which is based on his earlier method [Asc08]. OCCULT utilizes the local loop directivity and the curvature radius constraints in coronal loop tracing. In particular, the curvature radius constraint enables better loop tracing by providing estimates of loop direction range based on previously-traced loop segment.

Other coronal loop segmentation works include the solar loop mining system by Durak et al. [Dur09, Dur10]. Their system includes a block-by-block loop detection processing to retrieve images with coronal loops from large number image datasets. Another recent coronal loop segmentation method is McAteer et al.'s *2D Wavelet-Transform Modulus Maxima Method* [McA10]. Their method uses the derivative of a 2D Wavelet-based smoothing function as an edge detector in segmenting coronal loop structures.

3 NEW APPROACH

Next, the new snake-based approach for automated segmentation of coronal loop structure is introduced.

The approach introduced in this paper exploits the Gaussian-shape of the coronal loop's cross-sectional intensity profile in its snake minimization process. In particular, the new energy terms based on this intensity profile property are considered to enable the snake to more-accurately lock on the coronal loop structures. We will call this new snake the *Gaussian-snake* (*G-snake*) in this paper. In our approach, a G-snake is positioned on a part of a coronal loop structure initially and "grows" along the coronal loop structure until the entire loop structure is segmented. Then, this process is repeated until all the coronal loop structures are considered.

Next, the details of the new approach including the pre-processing steps to "clean" the image and the initialization and minimization of the G-snake are discussed.

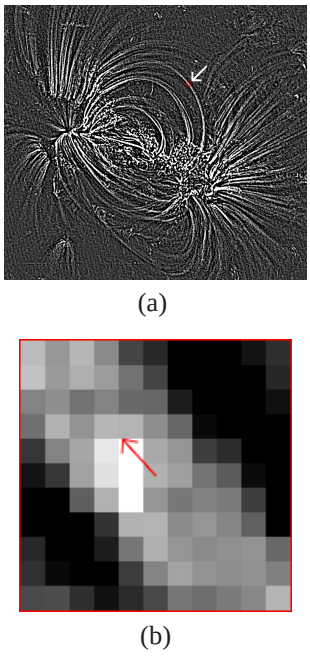
3.1 Pre-processing Steps

In our approach, a series of pre-processing steps are applied to the coronal image to remove image noise (e.g., impulse noise) and to enhance the contrast between the coronal loops and the background; thus, these steps provide images that are well-suited for the G-snake to segment the coronal loops accurately.

To remove impulse noise, 3×3 median filtering and 3×3 low-pass filtering are applied. Then, to sharpen the loop structures, a 3×3 unsharp masking is applied. (Unsharp masking subtracts a blurred version of an image from the image itself to reduce image intensity by a local background defined by the smoothing process [Gon02].) We note that the pre-processing steps used here were empirically determined and are similar to the ones used in [Lee06, Lee06b, Asc10]. Figure 1 (b) shows the pre-processed image of the coronal image shown in Figure 1 (a). As shown in the figure, the pre-processed image contains less noise and the coronal loop structures are much more distinguishable.

3.2 G-snake Initialization

The first step of the G-snake approach is to position a G-snake on a coronal loop structure automatically. (Here, we note that automated initialization of snake position is a very challenging task for any type of snake-based approaches.)



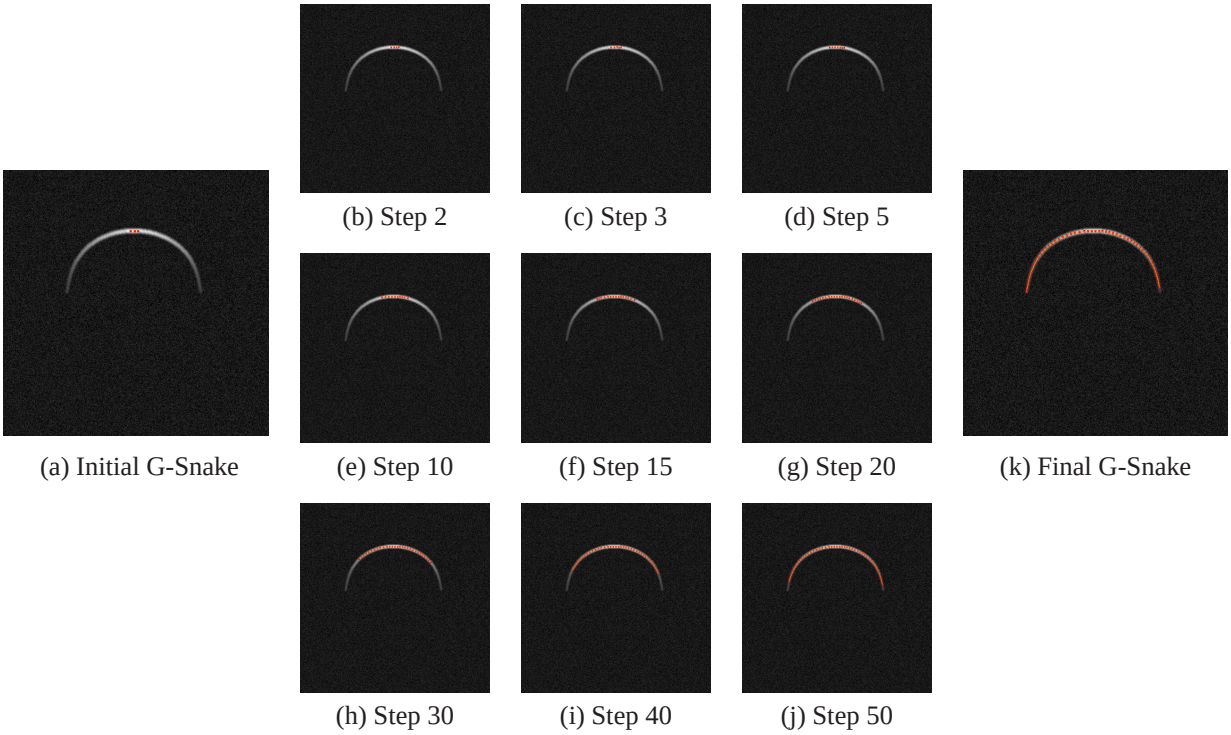
(a)

(b)

Figure 2: Directionality of a loop point (center point of the box) determined by PCA

In our approach, a set of initial snake points are positioned on the image region where the points in the region have very similar angular directions (since the nearby points on the same coronal loop have similar angular directions). In this paper, we denote the angular direction of a point as the *directionality* of the point and defined as the angular direction of high-intensity point variation in a small image region around the point. The directionality is measured using the principal component analysis (PCA) as the arctangent of the maximum eigenvector's components. (Here, we note that we empirically determined that 11×11 image region is suitable image size to compute the directionality of a point.) Figure 2 shows an example of directionality of a loop point determined using PCA. The directionality of the loop point (i.e., the center point of the red box shown in Figure 2 (a)) determined using PCA is marked as a red arrow in Figure 2 (b). As shown in the figure, PCA determines the loop direction accurately. (We note that this directionality measure has previously been used in [Lee06b].)

Using the directionality measures, we search for $N_1 \times N_2$ rectangular image regions where the directionalities of the points in the region are very similar. (We used $N_1 = 10$ and $N_2 = 3$ for the G-snake.) Specifically, we place a rectangular box on the image where the longer side of the box is parallel to the x-axis of the image. Then, we rotate the rectangular box with



(a) Initial G-Snake (b) Step 2 (c) Step 3 (d) Step 5 (e) Step 10 (f) Step 15 (g) Step 20 (k) Final G-Snake (h) Step 30 (i) Step 40 (j) Step 50

Figure 3: Illustration of G-Snake minimization

respect to the region center point's directionality and determine the divergence of the directionalities for the points in the rotated rectangular box. If the directionality divergence of the rectangular box is smaller than a threshold value, three equally-spaced G-snake points are positioned along the center of the rectangular image region. We empirically determined that three equally-spaced points are a good set of initial G-snake points as they are positioned closely such that they are always on the same coronal loop structure. An example of G-snake initialization is shown in Figure 3 (a). In the figure, three G-snake points (shown in red) are positioned on a synthetic loop structure successfully.

3.3 G-snake Energy Minimization

For each initial G-snake points, the G-snake energy function is minimized to segment the coronal loop structure.

Our G-snake uses the greedy-based algorithm in its minimization process. In the greedy-based algorithm, a G-snake is defined as a set of N points and each point is moved to a new position of the local minimum (which leads to the global minimization).

The energy function used in the G-snake are

$$\begin{aligned} E_{G-snake} = \sum_{i=1}^{N} & (a_i E_{cont} + b_i E_{curv} \\ & + c_i E_{grow} + d_i E_{img} \\ & + e_i E_{gauss} + f_i E_{direc}), \end{aligned} \tag{3}$$

where a_i, b_i, c_i, d_i, e_i, and f_i are the energy weights ($0 \leq a_i, b_i, c_i, d_i, e_i, f_i \leq 1$) and E_{cont}, E_{curv}, E_{grow}, E_{img}, E_{gauss}, and E_{direc} are the continuity energy, the curvature energy, the growth energy, the image energy, the Gaussian energy, and the directionality energy, respectively. (For the segmentation results shown in Section 4, an equal weight of 1 is used for all the energies.)

E_{cont} forces the distance between adjacent points to be maintained at an equal distance. E_{cont} is defined as

$$E_{cont} = (\overline{d} - \|v_i - v_{i-1}\|)^2, \tag{4}$$

where $\overline{d}$ is the average Euclidean distance between all adjacent G-snake points.

E_{curv} forces the snake to be smooth and avoid oscillations. E_{curv} is defined as

$$E_{curv} = \|(v_{i+1} - v_i) - (v_i - v_{i-1})\|^2. \tag{5}$$

E_{grow} is a new energy that forces the G-snake to grow along a coronal loop. To prevent "over-growing", E_{grow} is considered only for the first and last G-snake points (i.e., $c_i = 0$ for all $i \neq 1$ and N). E_{grow} is defined as

$$E_{grow} = -\|v_i - v_{i-1}\|^2. \tag{6}$$

E_{img} forces the G-snake points to move toward points with high intensity values. E_{img} is defined as

$$E_{img} = -I(x, y), \tag{7}$$

where $I(x, y)$ is the intensity value at a point (x, y).

While E_{img} attracts the G-snake points to points on the coronal loop structures (as the coronal loop appear

very bright in the image), it can also move the G-snake toward noise pixels with high intensity. To prevent this effect, new energies, E_{gauss} and E_{direc}, are used in G-snake.

E_{gauss} forces the G-snake to move toward the points whose intensity profile along the direction that is perpendicular to the loop direction is well-fitted by a Gaussian curve. (As mentioned in Section 2, the cross-section intensity profile of a coronal loop exhibits a Gaussian-shape.) E_{gauss} is defined as

$$E_{gauss} = -G(A,B), \tag{8}$$

where A is the height (i.e., peak intensity) and B is the center (i.e., horizontal position of peak) of the Gaussian that has been fitted using the intensity profile along the direction that is perpendicular to the loop direction. G is defined as

$$G(A,B) = \begin{cases} 0, & \text{if } \textit{poor Gaussian fit} \\ |B - C_p|, & \text{otherwise,} \end{cases} \tag{9}$$

where C_p is the center of the profile. We define a *poor Gaussian fit* case as a case when A is negative or too large or when the sum of squared Gaussian fitting error is too large. We have used the Levenberg-Marquardt algorithm [Lev44, Mar63] for the Gaussian curve fitting. (We note that we empirically determined that using 10 image points produces reasonable fitting results for the TRACE images).

E_{direc} forces the G-snake to move toward the points whose directionality is consistent with nearby loop points' directionalities. E_{direc} constraints the snake points to stay on one loop structure– it prevents a snake point to move toward other nearby loops. E_{direc} is defined as

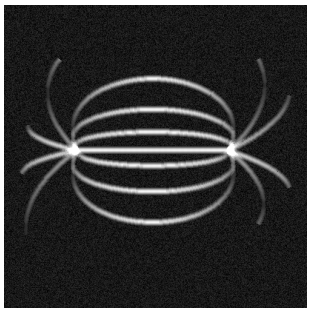
$$E_{direc} = \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} |d_{ij} - \mu|, \tag{10}$$

where d_{ij} denotes the directionality of a point at (i, j) and μ denotes the mean directionality in a $N_a \times N_b$ image region.

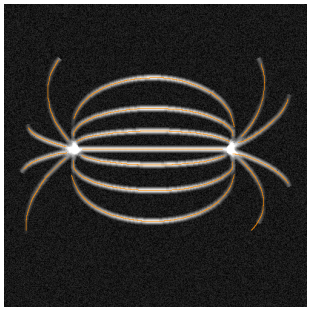
$E_{G-snake}$ is minimized iteratively until the G-snake becomes *stable*. We define a *stable* G-snake as a G-snake whose the number of points moving to new position is zero or the number of times points oscillating the same points is more than a pre-defined threshold value.

Here, we note that using only three initial G-snake points is often not enough to segment a (long) coronal loop structure. Thus, in our approach, an intermediate point is added to each G-snake when the distance between two adjacent G-snake points is greater than a pre-defined threshold (e.g., 10 in pixels).

Figure 3 shows an example of G-snake minimization process for a synthetic loop. As shown in the figure, the G-snake grew toward the ends of the loop and segmented the loop successfully. (Here, we note that the snake without the new energy terms can not segment the loop correctly; the new energy terms introduced here enable segmentation of loop structures that exhibit the Gaussian-shaped cross-sectional intensity profile.)



(a)



(b)

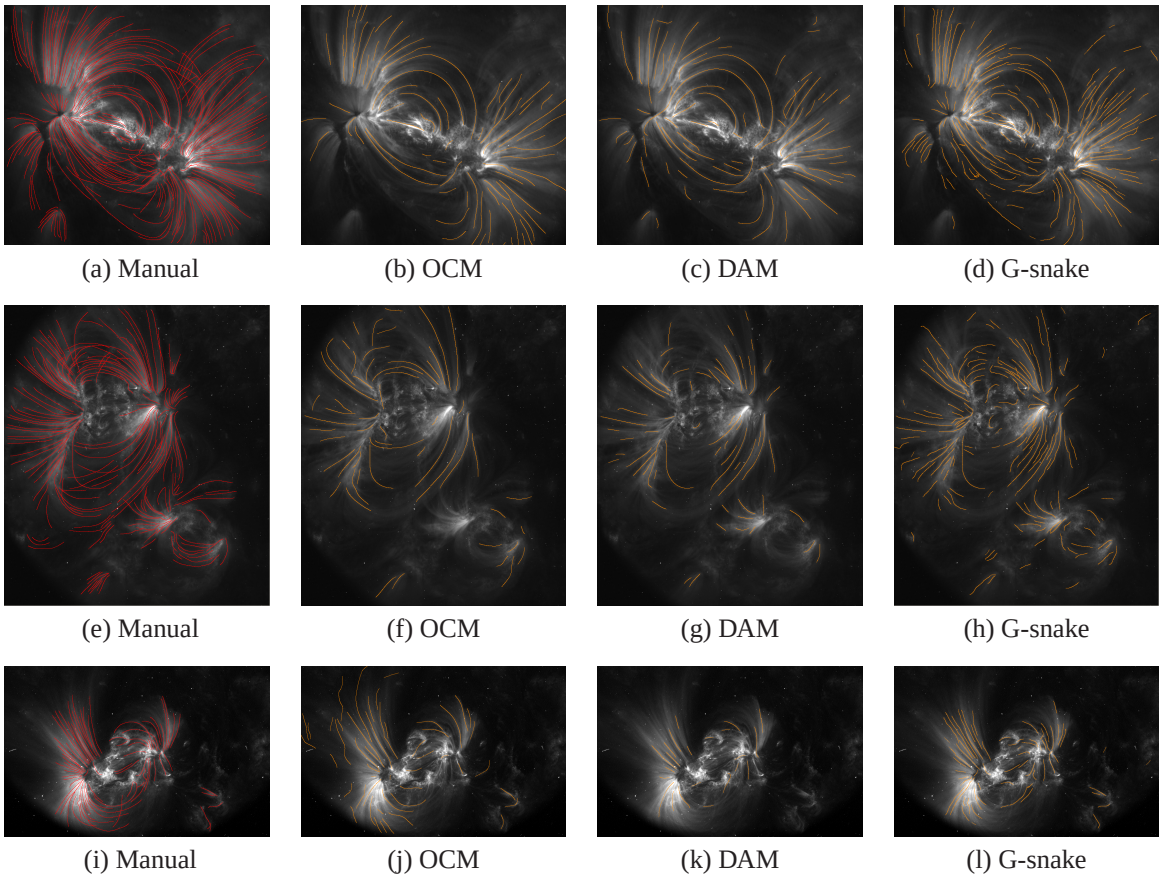
Figure 4: (a) Synthetic image and (b) Segmented loops

4 EXPERIMENTAL RESULTS

We have applied our new method to synthetic and real coronal images. In this paper, we report segmentation results for ten synthetic and five real images. We have used a similar scheme used in [Lee06] to generate the synthetic coronal images. Specifically, a simple magnetic field model is used to create "loop lines" on 1024×1024 images and these loop lines simulated to follow Gaussian-shaped cross-sectional intensity profiles. In addition, normal-distributed random image noise is added to simulate real coronal image noise. The

Table 1: GPE measures on synthetic images

Datasets	Max.	Min.	Mean	Std. dev.
Syn. 1	18.44	0.00	0.32	1.05
Syn. 2	11.18	0.00	0.30	0.76
Syn. 3	7.07	0.00	0.57	1.01
Syn. 4	3.00	0.00	0.27	0.46
Syn. 5	6.74	0.00	0.35	0.59
Syn. 6	24.17	0.00	0.45	1.00
Syn. 7	12.81	0.00	0.33	0.60
Syn. 8	2.24	0.00	0.28	0.46
Syn. 9	4.00	0.00	0.29	0.48
Syn. 10	2.83	0.00	0.34	0.49
Average	9.24	0.00	**0.35**	0.69



(a) Manual (b) OCM (c) DAM (d) G-snake

(e) Manual (f) OCM (g) DAM (h) G-snake

(i) Manual (j) OCM (k) DAM (l) G-snake

Figure 5: Coronal loop segmentation on real images (OCM and DAM results are borrowed from [Lee06, Lee06b])

size of the tested real images was 1024×1024. (Here, we note that different set of G-snake's parameters might be needed for non-TRACE coronal images.) To evaluate the effectiveness of G-snake, we have measured the four metrics (i.e., maximum, minimum, mean, and standard deviation) of the global positional error (GPE). GPE is defined as the global estimation of the shortest Euclidean distance difference between the real loops and the segmented loops [Lee06].

Figure 4 shows the G-snake segmentation result for one synthetic image. As shown in the figure, G-snake well-segmented the loop structures.

Table 1 shows the four metrics of GPE measures for the ten synthetic images we tested. For these synthetic images, the average of the mean GPE was only 0.35 (in pixels).

For the real coronal image testings, we have used the manual segmentation as the gold standard (since the actual loop positions were unknown). In addition, we have compared the G-snake's results with the OCM and DAM results. Figure 5 shows the segmentation results on three real coronal images. In the figure, the manually-segmented loops are overlaid with red curves (in the sub-figures (a), (e), (i)) and the automatically-segmented loops are overlaid with orange curves. Sub-figures (b), (f), and (j) show the OCM results, Sub-figures (c), (g), and (k) show the DAM results, and Sub-figures (d), (h), and (l) show the G-snake results. As shown in the figure, the G-snake seems to produce the best coronal loop segmentation results; the G-snake segmented many of the loops that were not segmented by OCM and DAM.

Table 2 shows the four metrics of GPE measures for OCM, DAM, and G-snake on real coronal images. We have chosen these five images for our testing as they have been used by others. As shown in the table, G-snake produced the minimum mean GPE measures for all images. The overall average of the mean GPE for G-snake was less than 1 pixel. (In some cases, the maximum and standard deviation GPE measures of G-snake were slightly higher than that of DAM.)

5 CONCLUSION

In this paper, the G-snake for automated coronal loop segmentation was described. The approach is the first method that utilizes a snake in coronal loop segmentation. It uses new energies (i.e., E_{grow}, E_{gauss}, and E_{direc}) in the snake's minimization process to enable accurate coronal loop segmentation. Through evaluation of the technique, we have shown that the G-snake can provide

Table 2: GPE measures for OCM, DAM, and G-snake on real coronal images

Datasets	Max.	Min.	Mean	Std. dev.
Real 1	34.79	0.00	2.19	3.21
Real 2	17.03	0.00	1.93	1.78
Real 3	47.41	0.00	2.82	4.71
Real 4	37.54	0.00	3.00	4.60
Real 5	57.01	0.00	3.11	5.94
Average	38.76	0.00	**2.61**	4.05

(a) OCM

Datasets	Max.	Min.	Mean	Std. dev.
Real 1	19.00	0.00	1.73	1.79
Real 2	11.18	0.00	1.67	1.39
Real 3	12.00	0.00	1.41	1.60
Real 4	36.35	0.00	2.36	3.64
Real 5	16.97	0.00	1.56	1.80
Average	19.10	0.00	**1.75**	2.04

(b) DAM

Datasets	Max.	Min.	Mean	Std. dev.
Real 1	24.19	0.00	0.85	1.39
Real 2	28.79	0.00	1.10	1.76
Real 3	13.60	0.00	0.78	1.05
Real 4	20.62	0.00	1.03	1.83
Real 5	27.20	0.00	1.07	1.75
Average	22.88	0.00	**0.97**	1.56

(c) G-Snake

consistent and accurate segmentation results of coronal loop structures.

In the future, we plan to perform further effectiveness comparisons with other coronal loop segmentation results (e.g., [Asc10]). In addition, we plan to adopt the G-snake-based coronal loop segmentation to other solar studies, such as solar magnetic field parameter recovery [Lee09]. We also hope to apply the G-snake to other scientifically-interesting structures that follows similar characteristics (i.e., G-snake may be adopted to segment other image structures that follow a different cross-sectional intensity profiles).

ACKNOWLEDGMENT

We acknowledge that the results of OCM and DAM were borrowed from [Lee06, Lee06b]. We also thank the reviewers for their valuable comments which improved our paper.

REFERENCES

[Ami90] Amini, A.A., Weymouth, T.E., and Jain, R.C., Using Dynamic Programming for Solving Variational Problems in Vision, Pattern Recognition, Vol. 12 (9), pp. 855–867, 1990.

[Asc08] Aschwanden, M.J., Lee, J.K., Gary, G.A., Smith, M., and Inhester, B., Comparison of Five Numerical Codes for Automated Tracing of Coronal Loops, Solar Physics, Vol. 248, pp. 359–377, 2008.

[Asc10] Aschwanden, M.J., A Code for Automated Tracing of Coronal Loops Approaching Visual Perception, Solar Physics, Vol. 262 (2), pp. 399–423, 2010.

[Cao09] Cao, C., Newman, T.S., and Germany, G.A., New Shape-based Auroral Oval Segmentation Driven by LLS-RHT, Pattern Recognition, Vol. 42 (5), pp. 607–618, 2009.

[Car03] Carcedo, L., Brown, D., Hood, A., Neukirch, T., and Wiegelmann, T., A Quantitative Method to Optimise Magnetic Field Line Fitting of Observed Coronal Loops, Solar Physics, Vol. 218, pp. 29–40, 2003.

[Dud00] Duda, R.O., Hart, P.E., and Strok, D.G., Pattern Classification, 2nd Edition, Wiley, New York, 2000.

[Dur09] Durak, N., Nasraoui, O., and Schmelz J., Coronal Loop Detection from Solar Images, Pattern Recognition, Vol. 42 (11), pp. 2481–2491, 2009.

[Dur10] Durak, N., Nasraoui, O., and Schmelz J., Automated Coronal-Loop Detection based on Cuntour Extraction and Contour Classification from the SOHO/EIT Images, Solar Physics, Vol. 264, pp. 383–402, 2010.

[Gon02] Gonzalez, R.C. and Woods, R.E., Digital Image Processing, 2nd Edition, Prentice Hall, Upper Saddle River, New Jersey, 2002.

[Hou62] Hough, P.V.C, Method and Means for Recognizing Complex Patterns, U.S. Patent, 3,069,654, 1962.

[Kas87] Kass, M., Witkin, A., and Terzopoulos, D., Snakes: Active Contour Models, International Journal of Computer Vision, Vol. 1 (4), pp. 321–331, 1987.

[Kuk04] Kukharev, G. and Nowosielski, A., Visitor Identification- Elaborating Real Time Face Recognition System, The 12th Int'l Conference in Central Europe on Computer Graphics, Visualization, and Computer Vision (WSCG 2004), Plzen, Czech Republic, pp. 157–164, 2004.

[Kul90] Kultanen, P., Xu, L., and Oja, E., Randomized Hough Transform (RHT), Proceedings, 10th International Conference on Pattern Recognition, Atlantic City, pp. 631–635, 1990.

[Lee09] Lee, J.K. and Gary, G.A., Recovery of

3D Solar Magnetic Field Model Parameter using Image Structure Matching, Proceedings, 4th International Conference on Computer Vision/Computer Graphics Collaboration Techniques, MIRAGE 2009, France, pp. 172-181, 2009.

[Lee06] Lee, J.K., Newman, T.S., and Gary, G.A., Oriented Connectivity-based Method for Segmenting Solar Loops, Pattern Recognition, Vol. 39 (2), pp. 246–259, 2006.

[Lee06b] Lee, J.K., Newman, T.S., and Gary, G.A., Dynamic Aperture-based Solar Loop Segmentation, Proceedings, 7th IEEE Southwest Symposium on Image Analysis and Interpretation, Denver, pp. 91–94, 2006.

[Lev44] Levenberg, K., A Method for the Solution of Certain Non-linear Problems in Least Squares, Quarterly of Applied Mathematics, Vol. 2, pp. 164–168, 1944.

[Mag09] Magalhaes, F., Oliveira, H.P., and Campilho, A.C., A New Method for the Detection of Singular Points in Fingerprint Images, Proceedings, IEEE Workshop on Applications of Computer Vision (WACV 2009), Snowbird, UT, pp. 1–6, 2009.

[Mar63] Marquardt, D.W., An Algorithm for Least Squares Estimation of Nonlinear Parameters, Journal of the Society for Industrial and Applied Mathematics, Vol. 11 (2), pp. 431–441, 1963.

[McA10] McAteer, R.T.J., Kestener, P., Arneodo, A., and Khalil, A., Automated Detection of Coronal Loops using a Wavelet Transform Modulus Maxima Method, Solar Physics, Vol. 262 (2), pp. 387–397, 2010.

[Pan01] Pankanti, S., Prabhakar, S., and Jain, A.K., On the Individuality Fingerprints, Proceedings, 2001 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001), Kauai, HI, pp. I-805–I-812, 2001.

[Sch99] Schrijver, C.J., Title, A.M., Berger, T.E., Fletcher, L., Hurlburt, N.E., Nightingale, R.W., Shine, R.A., Tarbell, T.D., Wolfson, J., Golub, L., Bookbinder, J.A., DeLuca, E.E., McMullen, R.A., Warren, H.P., Kankelborg, C.C., Handy, B.N., and Pontieu, B.D., A New View of the Solar Outer Atmosphere by the Transition Region and Coronal Explorer, Solar Physics, Vol. 187, pp. 261–302, 1999.

[Sel08] Sellah, S. and Nasraoui, O., An Incremental Hough Transform for Detecting Ellipses in Image Data Streams, Proceedings, IEEE International Conference on Tools with Artificial Intelligence, Dayton, Ohio, pp. 45–48, 2008.

[Smi05] Smith, M., 2005, http://twiki.mssl.ucl.ac.uk/twiki/bin/view/SDO/LoopRecognition, accessed on May 16, 2010.

[Ste98] Steger, C., An Unbiased Detector of Curvilinear Structures, IEEE Trans. on Pattern Analysis and Machine Intelligenc, Vol. 20 (2), pp. 113–125, 1998.

[Teo05] Teo, C.C. and Ewe, H.T., An Efficient One-Dimensional Fractal Analysis for Iris Recognition, The 13th Int'l Conference in Central Europe on Computer Graphics, Visualization, and Computer Vision (WSCG 2005), Plzen, Czech Republic, pp. 157–160, 2005.

[Wil92] Williams, D.J. and Shah, M., A Fast Algorithm for Active Contours and Curvature Estimation, Computer Vision, Graphics and Image Processing, Vol. 55 (1), pp. 14–26, 1992.

[Wis97] Wiskott, L. and Fellous, J.M. and Kruger, N. and Malsburg, C.V.D., Face Recognition by Elastic Bunch Graph Matching, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19 (7), pp. 775–779, 1997.