

# A Code for Automated Tracing of Coronal Loops Approaching Visual Perception

Markus J. Aschwanden

Received: 21 October 2009 / Accepted: 17 February 2010 / Published online: 10 March 2010  
© Springer Science+Business Media B.V. 2010

**Abstract** We develop a new numerical code with automated feature extraction, customized for tracing of coronal loops, a method we call *Oriented Coronal CURved Loop Tracing* (OCCULT), which for the first time breaks even with the results of visual tracing. The method used is based on oriented-directivity tracing of curvi-linear features, but in contrast to other general feature-extraction algorithms, it is customized for solar EUV and SXR images by taking advantage of the specific property that coronal loops have large curvature radii compared with their widths. We evaluate the performance of this new code by comparing the cumulative distribution of loop lengths, the median and maximum loop lengths, the completeness of detection, and the congruency of the detected features with other numerical codes and visual tracings. We find that the new code closely approaches the results of visual perception and outperforms the other existing numerical codes. This algorithm is useful for the 3D reconstruction of the geometry, motion, and oscillations of coronal loops, with single or stereoscopic spacecraft, as well as for modeling of the loop hydrodynamics and the coronal magnetic field.

**Keywords** Sun: EUV · Sun: magnetic fields · Methods: pattern recognition

## 1. Introduction

The rapidly growing archives of solar imaging data produced by extreme-ultraviolet (EUV) and soft X-ray (SXR) telescopes, such as from the SOHO, TRACE, STEREO, *Hinode*, and SDO space missions, put increasing pressure on scientists to develop and pursue automated data-analysis tools. However, meaningful and accurate data analysis can only be performed

---

Solar Image Processing and Analysis  
Guest Editors: J. Ireland and C.A. Young

M.J. Aschwanden (✉)

Solar and Astrophysics Laboratory, Lockheed Martin Advanced Technology Center, Dept. ADBS,  
Bldg. 252, 3251 Hanover St., Palo Alto, CA 94304, USA  
e-mail: [aschwanden@lmsal.com](mailto:aschwanden@lmsal.com)

if the selection and definition of events and phenomena is discriminant and self-consistent, which requires efficient feature-extraction tools as a first step.

In this study we focus on the automated recognition of coronal loops in EUV and SXR images. Some general introductions into pattern and feature detection in digital images can be found in the textbooks of Gonzalez and Woods (2008), Jähne (2005), Ripley (1996), or Pao (1996). Recent reviews specialized on image-processing techniques and feature recognition in solar physics are given in Aschwanden (2005, 2010), where feature detection in 2D images covers curvi-linear and region-based features, spectral methods, and artificial intelligence applications.

The new method of automated detection of coronal loops that we develop and describe in this study has some heritage from earlier numerical codes, whose description and comparison of performance are documented in the study of Aschwanden *et al.* (2008a). The five numerical codes compared therein include:

- i) an early oriented-directivity code that uses external information from a magnetic field extrapolation (Lee, Newman, and Gary, 2004, 2006a);
- ii) a dynamic aperture-based method of solar-loop segmentation (Lee, Newman, and Gary, 2006b);
- iii) an unbiased detection method of curvi-linear structures developed by Steger (1996), Raghupathy (2004), and Smith (2005);
- iv) an oriented-directivity code developed by Aschwanden; and
- v) a code for ridge detection by automated scaling (Inhester, Feng, and Wiegelmann, 2007).

More recently, new developments have been explored with

- vi) the detection of elliptical loop geometries using the incremental Hough transform (Selah and Nasraoui, 2008);
- vii) supervised (neural) learning strategies based on classifiers, such as RIPPER, Multi-Layer Perceptron, or Adaboost (Durak, Nasraoui, and Schmelz, 2009, 2010), and
- viii) a wavelet-transform modulus maxima method (McAteer *et al.*, 2010).

A number of these codes employ general pattern-recognition methods, such as image segmentation, Radon transform, wavelet transform, Hough transform, ridgelet detection, or neural learning, while others are geared more towards specific patterns of coronal loops, which follow the magnetic field and thus have elliptical shapes, large curvature radii, Gaussian cross-section profiles, and can be traced with oriented-directivity methods. A list of existing automated loop tracing codes is provided in Table 1. Our new code is even more customized for detection of coronal loops, containing a few tuning parameters that have been optimized and trained by visual tracing of coronal loops in a set of high-resolution EUV images from TRACE. We find that the performance of the new code closely matches the results of visual perception, so that it can be considered as the first achieved break-even with human pattern recognition. The training of the tuning parameters with feedback from visual tracing could be further developed with neural learning strategies.

In Section 2 we describe the method of the automated detection code, in Section 3 we present performance tests on a standard image that has been used in the comparison of numerical detection codes earlier (Aschwanden *et al.*, 2008a), in Section 4 we show four more test cases with performance results, and in Section 5 we briefly discuss the salient features and caveats of the new method, as well as future applications.

**Table 1** Automated loop-tracing algorithms used in solar images.

Feature Detection Algorithm	Name	Reference
i) Oriented-Connectivity Method	OCM	Lee, Newman, and Gary (2004, 2006b)
ii) Dynamic Aperture-based loop segmentation Method	DAM	Lee, Newman, and Gary (2006b)
iii) Unbiased Detection of curvi-linear structures	UDM	Smith (2005)
iv) Oriented-Directivity loop tracing Method	ODM	Aschwanden (2008)
v) Ridge detection by Automated Scaling	RAS	Inhester, Feng, and Wiegmann (2007)
vi) Ellipse detection, inverse Hough transform	...	Sellah and Nasraoui (2008)
vii) Supervised neural learning strategies with classifiers	RIPPER	Durak, Nasraoui, and Schmelz (2009, 2010)
viii) Wavelet-transform moldula maxima method	...	McAteer <i>et al.</i> (2010)
ix) Oriented Coronal CURved Loop Tracing	OCCULT	This work

## 2. Method and Numeric Code

Our new code is a further development of the oriented-directivity loop tracing method (ODM) that was briefly described in Section 2.4 of Aschwanden *et al.* (2008a). The new code has some heritage, but also deviates significantly from the original code. The term “oriented-directivity” has the general meaning of guided tracing by some directivity criterion, which can be external information from magnetograms (Lee, Newman, and Gary, 2004, 2006a), or simply a predicted range based on the direction from the previously traced loop segment, which is essentially a curvature constraint. Since the basic feature of our new code is the oriented guiding by the loop curvature, we refer to the code with the acronym “Oriented Coronal CURved Loop Tracing” (OCCULT).

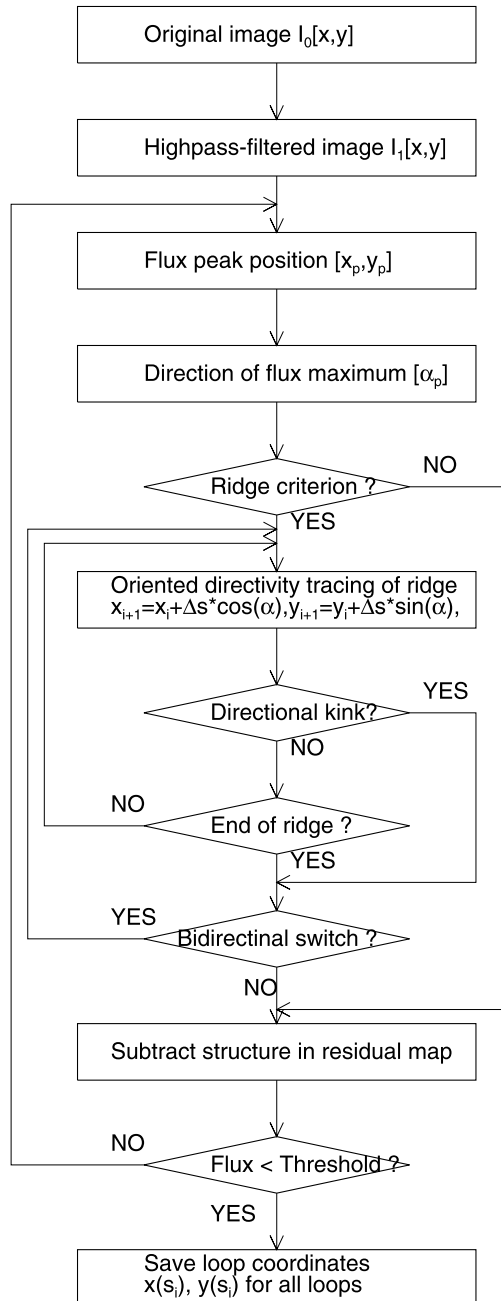
The code is made available for public use in an IDL-based software library of the *Solar Software* (SSW) (Freeland and Handy, 1998), with tutorials on the website <http://www.lmsal.com/~aschwand/software/>.

### 2.1. Flow Chart of the Numeric Code

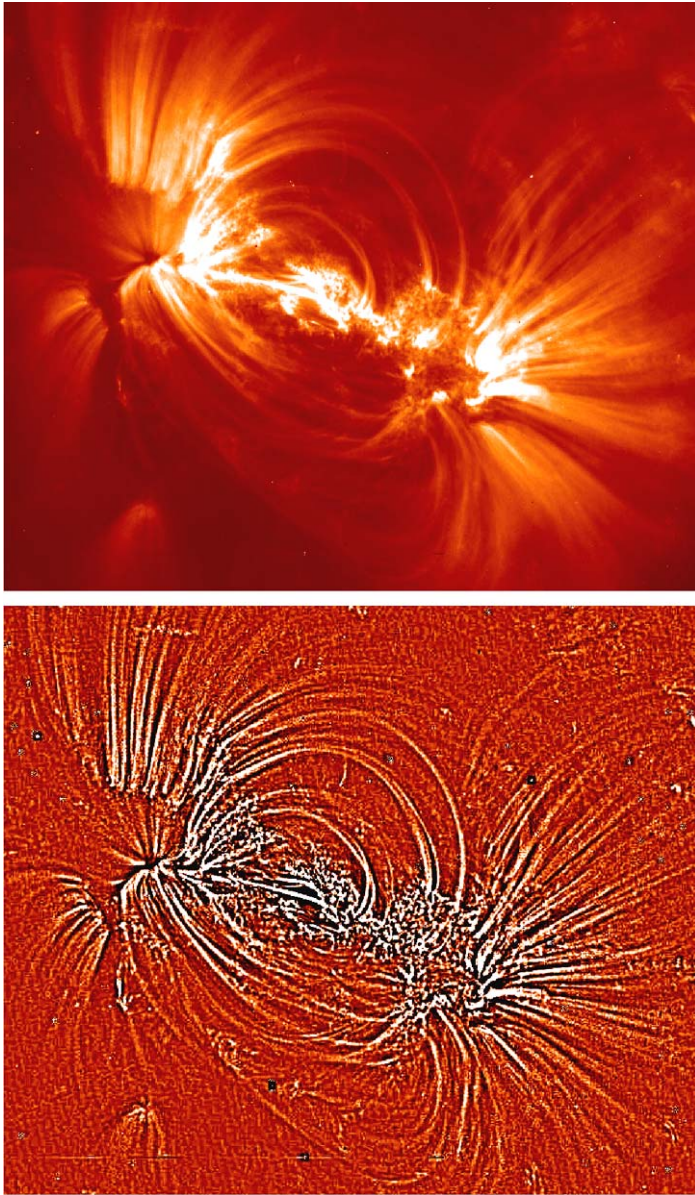
A schematic description of the numerical code is visualized as a flow chart in Figure 1. The input is a 2D digital image  $I(x_i, y_j)$ ,  $i = 1, \dots, N_x$ ,  $j = 1, \dots, N_y$  of observed flux intensities ( $I$ ). The dimensions of tested images (mostly from TRACE) range up to  $N_x = N_y = 1024$ , and the units of the observed EUV flux are typically datanumbers per second ( $\text{DN s}^{-1}$ ), but the code is designed for arbitrary flux units and refers to spatial scales in dimensionless pixel units.

The basic algorithm consists of an iterative detection of loop structures, starting from the brightest loop down to the faintest one above some noise threshold. The code picks up a structure at its brightest (peak) location  $(x_p, y_p)$ , decides whether the peak is part of a curvi-linear ridge or not, finds the directivity angle ( $\alpha$ ) of the ridge at the flux peak, proceeds in tracing the ridge in one direction according to an oriented-directivity criterion  $|\Delta\alpha| \leq \alpha_{\max}$ , detects one endpoint  $(x_0, y_0)$  of the loop, where the angular change exceeds the maximum allowable value ( $\alpha_{\max}$ ) or the flux vanishes, turns around to the peak  $(x_p, y_p)$  and pursues a bi-directional tracing in the opposite direction to the secondary endpoint  $(x_n, y_n)$ . The loop coordinates  $(x_i, y_i)$ ,  $i = 0, \dots, n$  are then regrouped in consecutive order from one endpoint of the loop to the other. The image area of the successfully traced loop, which is a curved stripe with length  $L_{\text{box}}$  and full width  $2w$ , is then erased from the highpass-filtered image, so

**Figure 1** Flow chart of automated loop-detection code, dubbed Oriented Coronal CURved Loop Tracing (OCCULT).



that the identified loop does not interfere with the iterative tracing of subsequent loops. Once the flux maximum of the residual image drops below the pre-defined noise threshold, the iteration stops and the entire set of traced loops is saved in the form of one-dimensional loop coordinate arrays  $x(s_i)$  and  $y(s_i)$  as a function of the loop-length coordinate  $s_i$ ,  $i = 1, \dots, n_s$



**Figure 2** Original image (top) and highpass-filtered image (bottom) observed with TRACE on 19 May 1998, 22:21 UT, in the 171 Å wavelength. The displayed parts cover the pixel ranges of  $i = 200, \dots, 1000$  ( $x$ -axis) and  $j = 150, \dots, 850$  ( $y$ -axis) of the original  $1024 \times 1024$  image, covering  $512''$ . The highpass-filtered image was created by subtraction of a smoothed image (with  $n_{\text{sm}} = 9$ ) from the original image. The color scale spans the flux range of three standard deviations from the average flux level for the original image, and one standard deviation for the highpass-filtered image, respectively.

for every loop. Some individual steps of this algorithm are described in more detail in the following subsections.

**Table 2** Fluxes and signal-to-noise ratio in a highpass-filtered TRACE image.

Smoothing box car $n_{\text{sm}}$	Flux standard deviation $S$ [DN s <sup>-1</sup> ]	Flux ratio $S/S_0$	Signal-to- noise ratio $S/N$	Ratio $(S/N)/(S_0/N_0)$
3	4.4	0.072	2.11	0.268
5	5.9	0.095	2.43	0.308
7	7.3	0.117	2.69	0.342
9	8.6	0.138	2.93	0.371
11	9.7	0.157	3.12	0.396
13	10.8	0.174	3.29	0.417
15	11.7	0.189	3.43	0.435
17	12.6	0.202	3.55	0.450
19	13.3	0.214	3.65	0.463
21	13.9	0.224	3.73	0.474
$\infty$	62.1	1.000	7.88	1.000

## 2.2. Highpass Filtering

The first step of the algorithm is a highpass filtering with a 2D box car of length  $n_{\text{sm}}$ , for which we simply choose a subtraction of a smoothed image (with box car  $n_{\text{sm}}$ ) from the original image  $I_0$  (related to the method of “unsharp masking”), yielding an image  $I_1$ :

$$I_1(x_i, y_j) = I_0(x_i, y_j) - \text{smooth}[I_0(x_i, y_j), n_{\text{sm}}]. \quad (1)$$

An example of an original image and the highpass-filtered image is shown in Figure 2, for the same standard image used in tests of automated detection codes earlier (Aschwanden *et al.*, 2008a), observed with the *Transition Region and Coronal Explorer* (TRACE) (Handy *et al.*, 1999) on 19 May 1998, 22:21 UT, in 171 Å, with a size of 1024 × 1024 pixels and a pixel size of 0.5″ ( $\approx 375$  km on the solar surface). Here we choose a smoothing box car of ( $n_{\text{sm}}$ ) nine pixels, which best enhances structures with a half width of ( $w$ ) about three pixels. The relation between the Gaussian half width ( $w$ ) of a loop cross-section and the highpass-filter constant that best enhances this structure is approximately,

$$n_{\text{sm}} = 3 + 2w, \quad (2)$$

so the smoothing constant is  $n_{\text{sm}} = 5, 7, 9, \dots$  for loops with half widths of  $w = 1, 2, 3, \dots$ . The minimum smoothing constant is  $n_{\text{sm}} = 3$ , which would enhance unresolved structures with a width of  $w \ll 1$ .

When choosing the smoothing constant  $n_{\text{sm}}$  in the highpass filtering, one has also to bear in mind that the spatial resolving power of the filter counterbalances the signal-to-noise ratio of the detected structures. We calculated the average and standard deviations of fluxes for the TRACE image shown in Figure 2 as a function of the highpass-filter smoothing constant ( $n_{\text{sm}}$ ) in Table 2. The standard deviation is a measure of the typical signal of faint loop structures, which varies from  $S = 4.4$  DN s<sup>-1</sup> (for  $n_{\text{sm}} = 3$ ) to  $S = 13.9$  DN s<sup>-1</sup> (for  $n_{\text{sm}} = 21$ ), compared with  $S = 62.1$  DN s<sup>-1</sup> in the unfiltered (original image), so the signal drops down to  $S/S_0 \approx 7.2\%$  for the finest filter ( $n_{\text{sm}} = 3$ ). The corresponding signal-to-noise ratio  $(S/N) \approx \sqrt{S}$  (using Poisson statistics and the fact that the flux unit of 1 DN s<sup>-1</sup> corresponds approximately to one incident photon for the TRACE telescope), drops from

$(S_0/N_0) = 7.88$  in the original image down to  $S/N = 2.11$  for the finest filter ( $n_{\text{sm}} = 3$ ). So, the faintest loops have a signal of about two standard deviations for the finest filter ( $n_{\text{sm}} = 3$ ), but stand out with about three standard deviations for  $n_{\text{sm}} = 9$ , which we choose here.

### 2.3. Noise Threshold Selection

For the automated detection of loop structures we design an iterative procedure, which starts with the brightest loop first and proceeds to the next brightest, until the algorithm stops at the faintest loop with a brightness level above a pre-set flux threshold ( $I_{\text{th}}$ ). In principle, a data noise threshold could theoretically be calculated in terms of a combination of Poisson noise and instrumental effects, but there are so many instrumental noise effects (hot pixels, dark pixels, high-energy particle spikes, readout noise, compression noise, *etc.*) that it is easier to evaluate an empirical threshold suitable for feature extraction. Since the highpass-filtered image has an average flux value around zero and symmetric positive and negative flux differences relative to a smoothed surface, it is most convenient to calculate the mean and standard deviation of the highpass-filtered image ( $I_1$ ),

$$I_1 \pm \sigma_{I_1}, \quad (3)$$

and to define a threshold  $I_{\text{th}}$  in units of the standard deviation,

$$I_{\text{th}} = I_1 + N_\sigma \sigma_{I_1}. \quad (4)$$

From tests with a number of TRACE images we find that a threshold value of  $N_\sigma \approx 1$ –2 yields a good discrimination between significant loop patterns and random noise structures. However, the noise varies somewhat in different images and the users should use their own judgment to choose a reasonable value for  $N_\sigma$ , depending on how sensitive the detection of loops is desired to be, without being contaminated by loop-unrelated random-noise structures. Choosing a significance level of  $N_\sigma = 1$  for the image shown in Figure 2 yields a signal of  $S = 8.6 \text{ DN s}^{-1}$  for the faintest loops using a highpass filter of width  $n_{\text{sm}} = 9$ , which has a signal-to-noise ratio of  $S/N = 2.9$  according to Table 2, so they have a sufficiently high significance.

### 2.4. Determination of Loop Directivity

The code starts to pick up a loop structure at its brightest position, which we denote as peak position  $(x_p, y_p)$ . Then the code determines the direction of the local ridge by sampling the average fluxes  $F_k$  in every direction  $\varphi_k$  within a radial distance  $\rho_l$  (over a length of  $\rho_l = (\rho_2 - \rho_1)$  pixels),

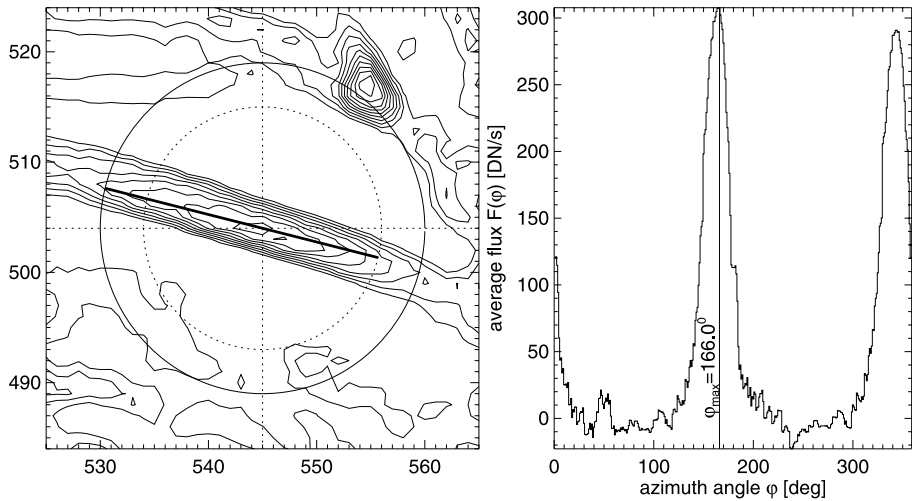
$$F_k = \frac{1}{\rho_l} \sum_{l=\rho_1}^{\rho_2} F(x_{kl}, y_{kl}), \quad (5)$$

with the fluxes  $F_{kl}$  sampled in the pixels next to the image positions

$$\begin{aligned} x_{kl} &= x_p + \rho_l \cos(\varphi_k), \\ y_{kl} &= y_p + \rho_l \sin(\varphi_k), \end{aligned} \quad (6)$$

where the radial limits  $\rho_1 = -(L_{\text{box}}/2 - \Delta s)$  and  $\rho_2 = (L_{\text{box}}/2)$  span the length  $L_{\text{box}}$  of a moving box, corrected for an offset of the moving step  $\Delta s$  in order to compensate the





**Figure 3** Local direction finding at the peak flux position of a ridge: A flux contour plot of the local neighborhood (x-pixel range = 525, ..., 565 and y-pixel range = 484, ..., 524, correspondingly) is shown (left frame) with the average flux  $[F(\phi)]$  sampled as a function of the azimuth angle (right frame), within a radius of  $r_1 = 15$  pixels in the forward direction and  $r_2 = 11$  pixels in the backward direction (circles in left frame). A maximum average flux is found for an azimuth angle of  $\phi_{\max} = 166^\circ$ , indicated with a thick straight line in the left frame.

curvature bias in the forward direction. With Equation (5) we obtain the average flux  $F_k$  in each direction  $\phi_k$  and can infer the direction  $\phi_{\text{dir}}$  of the local ridge from the maximum,

$$F_{\max} = \max[F_k(\phi_{\text{dir}} = \phi_k)]. \quad (7)$$

An example of this direction-finding algorithm is shown in Figure 3, where a contourplot of the ridge is shown (Figure 3 left) and the average flux  $F_k$  as a function of the directional angle  $\phi_k$ .

## 2.5. Loop Curvature Restriction

The underlying assumption in this procedure is that the ridge is essentially straight, or has a large curvature radius ( $r_{\text{curv}}$ ), compared with the box length ( $L_{\text{box}}$ ). Typically we choose a length of  $L_{\text{box}} = 30$  pixels. For the half width of detected loops we have chosen  $w \approx 3$ , corresponding to an optimum filter constant  $n_{\text{sm}} = 9$  according to Equation (2). Thus, the angle ( $\Delta\alpha$ ) of a loop segment with half length  $L_{\text{box}}/2$  and curvature radius  $r_{\text{curv}}$  is defined by (see Figure 4)

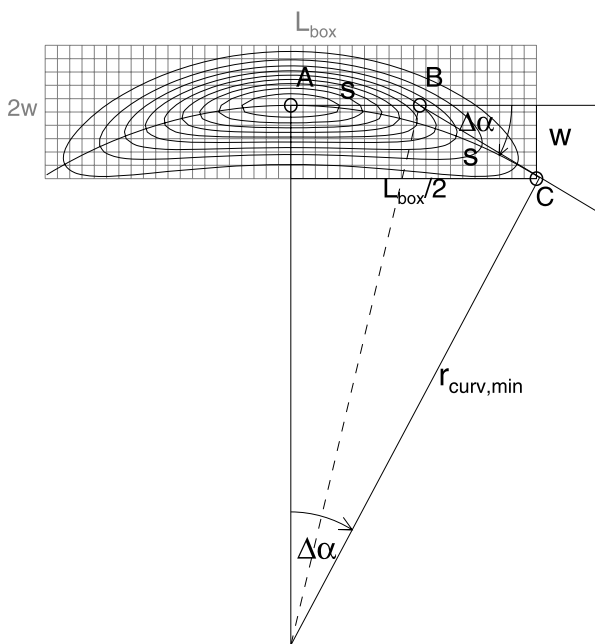
$$\sin(\Delta\alpha) = \frac{L_{\text{box}}/2}{r_{\text{curv}, \min}}. \quad (8)$$

The perpendicular deviation ( $w$ ) of the curved loop from a straight tangential line over a segment length  $L_{\text{box}}/2$  is given by (see triangle with hypotenuse  $BC$  in Figure 4)

$$\sin(\Delta\alpha) \approx \frac{w}{s}, \quad (9)$$



**Figure 4** Geometry of the loop-curvature restriction: In order to encompass a curved loop structure inside a box with length  $L_{\text{box}}$  and half width  $w$ , the curvature radius has to fulfill the relation  $r_{\text{curv},\text{min}} \gtrsim L_{\text{box}}^2/8w$ . See derivation in text, Equation (10).



where the length ( $s$ ) of the segment  $BC$  is approximately  $L_{\text{box}}/4$ . This leads, with Equations (8) and (9), to the relation

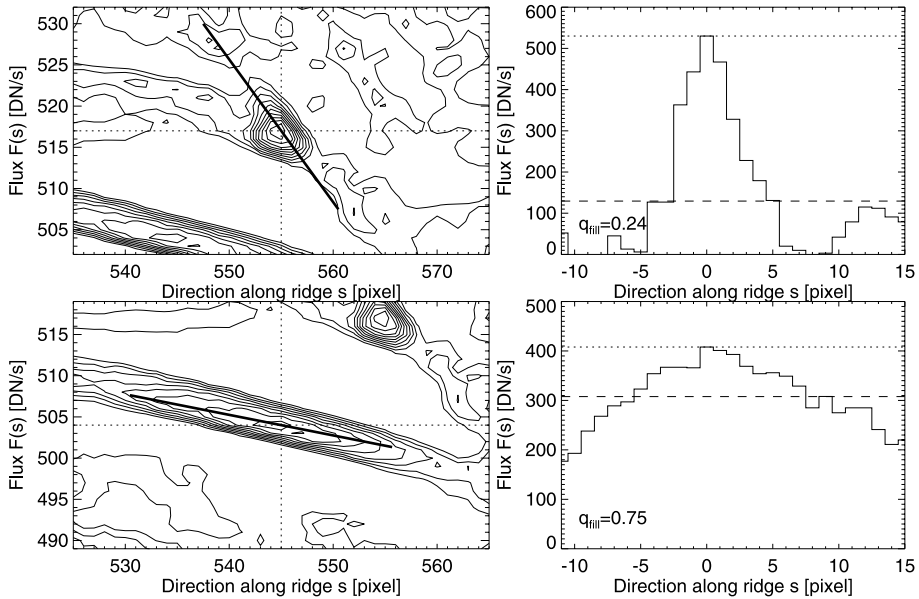
$$r_{\text{curv}} > r_{\text{curv},\text{min}} \approx \frac{L_{\text{box}}^2}{8w}. \quad (10)$$

For our chosen values of  $L_{\text{box}} \approx 30$  and  $w \approx 3$  we obtain  $r_{\text{curv}} \gtrsim 30^2/(8 \times 3) = 38$  pixels. For every loop with larger curvature radii we can trace the maximum flux along the ridge inside a straight box with length  $L_{\text{box}}$  and half width  $w$ . For slightly smaller curvature radii, the so-determined flux along the ridge has still a maximum along a straight direction, but for significantly more strongly curved loops, the ridge will curve outside of the box and other adjacent loops will confuse and detract the tracing of the primary loop.

Therefore, we expect an efficient detection of straight loops or loops with large curvature radii (compared with their width), which is satisfied for most coronal loops. A statistical study on loop widths and lengths observed with SOHO/EIT has shown an average loop length of  $L = 433 \pm 136$  Mm and a loop width of  $w = 7.1 \pm 0.8$  Mm (Aschwanden *et al.*, 1999a), which yields typical curvature radii of  $r_{\text{curv}} \approx (L/\pi) \approx 137 \pm 43$  Mm, being about a factor of 20 larger than the average loop width.

## 2.6. Ridge Criterion

Our algorithm picks up a loop structure at its brightest position, because our iterative detection and elimination scheme in the residual map proceeds from the brightest to the faintest pixel. However, the brightest pixel of a residual map is not always part of a coronal loop. There are other bright structures in EUV images, such as “moss structures” (Berger *et al.*, 1999), which are the transition regions of hotter loops whose coronal parts radiate in SXR but are invisible in EUV, nonsolar signals (high-energy particle spikes), or instrumental effects (hot pixels, pixel bleeding, *etc.*). In order to prevent tracing of structures starting with



**Figure 5** Examples of a loop-like ridge structures (bottom) and of a non-loop ridge structure (top). A contour map of the local neighborhood is shown (left panels), with the ridge peak (dotted cross-hair) and direction (thick line) indicated, as well as the flux profiles  $F(s)$  along the ridge direction (right panels). The ratio of the average flux  $F_{\text{avg}}$  (dashed line in right panel) to the peak flux (dotted line in right panel) defines the filling factor, i.e.  $q_{\text{fill}} = F_{\text{avg}}/F_{\text{peak}}$ , which discriminates between the two structures.

spurious intensity peaks and connecting to accidental neighboring loop structures we found it useful to define a “ridge criterion” that discriminates between ridge-like topologies that are typical for coronal loops and spike-like intensity peaks that are attributed to non-loop features. An efficient criterion to discriminate between elongated and spike-like structures is the ratio of their FWHM along the ridge axis to the length of the search box ( $L_{\text{box}}$ ), which we call the “filling factor” ( $q_{\text{fill}}$ ),

$$q_{\text{fill}} = \frac{\text{FWHM}}{L_{\text{box}}}. \quad (11)$$

For a straight ridge with near-constant flux along the ridge axis, this ratio is near unity. If there is a short spike with a length of FWHM and a peak flux of  $F_{\text{peak}}$ , the total flux is  $F_{\text{peak}} \times \text{FWHM}$ . Thus, we have the following relation for the total flux along the box axis with length  $L_{\text{box}}$ ,

$$F_{\text{total}} = \int F(s) ds = F_{\text{avg}} L_{\text{box}} = F_{\text{peak}} \text{FWHM}, \quad (12)$$

which allows us to measure the filling factor from the ratio of the average  $F_{\text{avg}}$  to the peak flux  $F_{\text{peak}}$ :

$$q_{\text{fill}} = \frac{\text{FWHM}}{L_{\text{box}}} = \frac{F_{\text{avg}}}{F_{\text{peak}}}. \quad (13)$$

From numerous tests with the goal of optimizing the automated detection to match the visual detection, we found a value of  $q_{\text{fill}} \gtrsim 0.25$  to be a suitable criterion for most images.

We show examples of a loop-like ridge structure and of a non-loop ridge structure in Figure 5 (both also visible in Figure 3), along with flux profiles  $[F(s)]$  along the ridge direction. The two structures have filling factors of  $q_{\text{fill}} = 0.75$  and  $q_{\text{fill}} = 0.24$ , so only the first one qualifies as a loop structure candidate, if we use the criterion of  $q_{\text{fill}} \geq 0.25$ .

## 2.7. Oriented-Directivity Tracing

After we identified a peak position  $(x_p, y_p)$ , determined the initial direction  $\varphi_{\text{dir}}$  of the ridge, and passed the ridge criterion (Equation (13)), we can start to trace a loop in one direction. We can proceed in one direction with a step of  $\Delta s$  that is as small as one pixel, or in larger steps as large as half a moving box, without losing track of the curved structure. We found a step size of  $\Delta s = 2w$ , but not larger than  $\Delta s \leq L_{\text{box}}/3$ , to be sufficient for most images. Starting from an old position  $(x_i, y_i)$  with direction  $\varphi_i$ , the predicted position of the next loop point  $(x_{i+1}, y_{i+1})$  is then

$$\begin{aligned} x_{i+1} &= x_i + \Delta s \cos(\varphi_i), \\ y_{i+1} &= y_i + \Delta s \sin(\varphi_i). \end{aligned} \quad (14)$$

To determine the exact position, allowing for a curvature change, we determine the direction  $\varphi_{i+1}$  of the flux maximum in the forward direction, within a cone of  $\varphi_i \pm \Delta\varphi$ , where  $\Delta\varphi \approx 30^\circ$  is suitable for most cases. Repeating the same procedure, we iterate until we reach a discontinuity that fails to pass either the curvature criterion (due to a large change in direction) or the ridge criterion (due to a large gap or peak). The algorithm then switches back and resumes the oriented-guided tracing in the opposite direction from the main peak  $(x_p, y_p)$ , until a second endpoint is reached, which terminates the tracing of one particular loop.

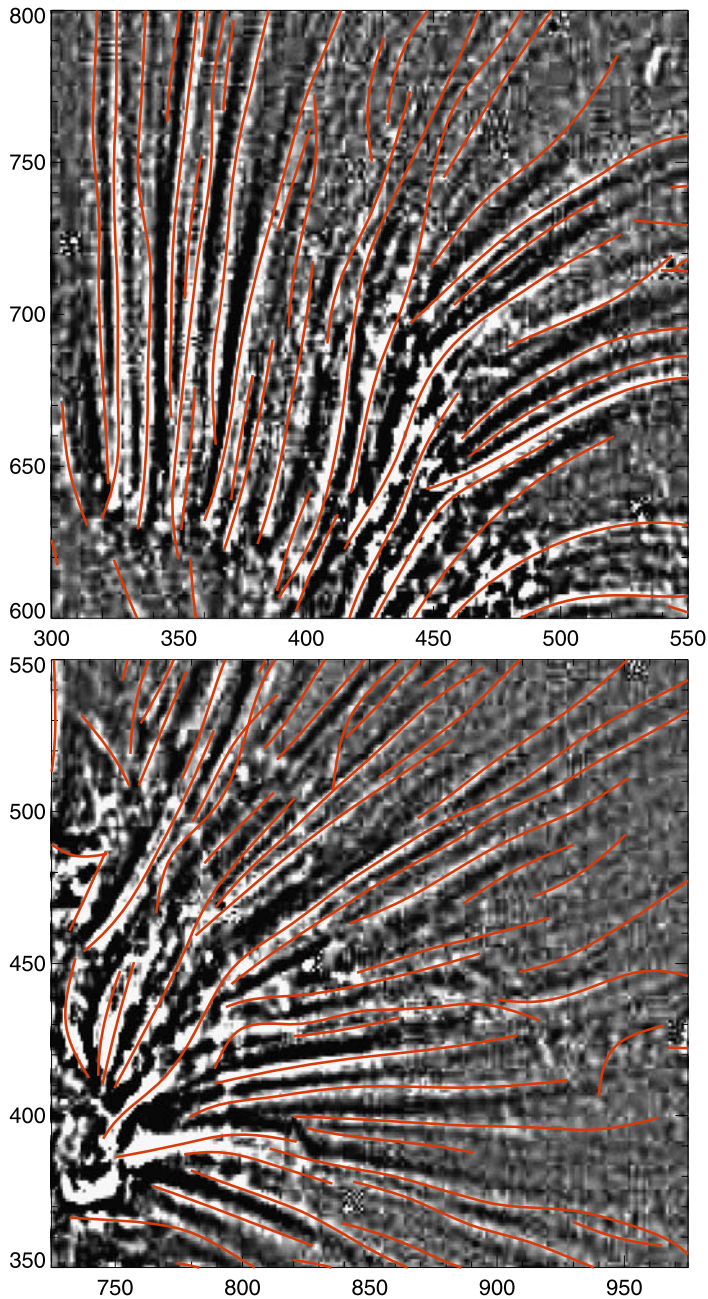
We show some details of loop tracings in two image sections with a size of  $250 \times 200$  pixels in Figure 6. The image sections are identical to Figures 7 and 8 in Aschwanden *et al.* (2008a), where it can be compared with the tracings of five previous numerical codes. The previous codes fail to detect about half of the loop structures in each image section, which are now completely detected with the new code according to a quick visual inspection.

## 2.8. Iterative Ridge Subtraction

The loop coordinates  $(x_i, y_i)$ ,  $i = 1, \dots, n_s$  of a completed loop tracing are represented by a curvi-linear 1D array. From this curved trajectory we calculate all pixel coordinates that have a distance of  $d \leq w$  from the loop “backbone” coordinates and set the residual image to zero at those (already traced) pixels. The elimination of this loop area from the residual image will guarantee that the same loop does not interfere with subsequent tracings, as well as that the maximum flux in the residual image diminishes. The algorithm then searches for the next maximum and continues to trace the next loop, until the maximum flux of the residual image after a number of loop tracings falls below the flux threshold (Figure 1). Finally, all loop coordinates are saved.

## 2.9. Tuning Parameters

In the automated loop-detection algorithm we reduced the important tuning parameters to three variables that can be chosen by the user, depending on the specific properties of the image or the topology of the desired structures: the length ( $L_{\text{box}}$ ) of the tracing box, the typical



**Figure 6** Details of loop tracings shown for two image sections of the original image shown in Figure 2. The image sections have the pixel ranges of  $x = 300, \dots, 550$ ,  $y = 600, \dots, 800$  (top panel), and  $x = 725, \dots, 975$ ,  $y = 350, \dots, 550$  (bottom panel), respectively. The image sections are identical to Figures 7 and 8 in Aschwanden *et al.* (2008a). The highpass-filtered image is shown in grayscale and the automated loop tracings with red curves.

half width ( $w$ ) of a loop structure, and the significance level ( $N_\sigma$ ) for the flux threshold in units of standard deviations of fluxes in the highpass-filtered image. The optimum smoothing constant ( $n_{\text{sm}}$ ) for the highpass filter is constrained by Equation (2). Additional tuning parameters of secondary importance are the filling factor ( $q_{\text{fill}}$ ) for the ridge criterion, the maximum cone angle ( $\Delta\varphi$ ) of directivity search, and the minimum acceptable loop length ( $L_{\text{min}}$ ). From comparisons of automated with visual tracings we established the following parameters to yield the best results in matching automated with visual detection, based on applications to EUV images from TRACE,

$$\begin{aligned}
 L_{\text{box}} &\approx 30 \text{ pixels}, \\
 w &\approx 3 \text{ pixels}, \\
 N_\sigma &\approx 1, \dots, 2, \\
 n_{\text{sm}} &= 3 + 2 w, \\
 q_{\text{fill}, \text{min}} &\approx 0.25, \\
 \Delta\varphi &\approx 30^\circ, \\
 L_{\text{min}} &\approx L_{\text{box}}.
 \end{aligned} \tag{15}$$

These control or tuning parameters can be selected or adjusted for every image individually. The code is available as software package in the IDL-based Solar Software (SSW), with tutorials and examples accessible on the website <http://www.lmsal.com/~aschwand/software/>.

### 3. Performance Tests

In this section we describe the test results of the automated loop-detection code applied to the TRACE 171 Å image of 19 May 1998, 22:21 UT, the same one used in performance comparisons of five numerical codes in Aschwanden *et al.* (2008a).

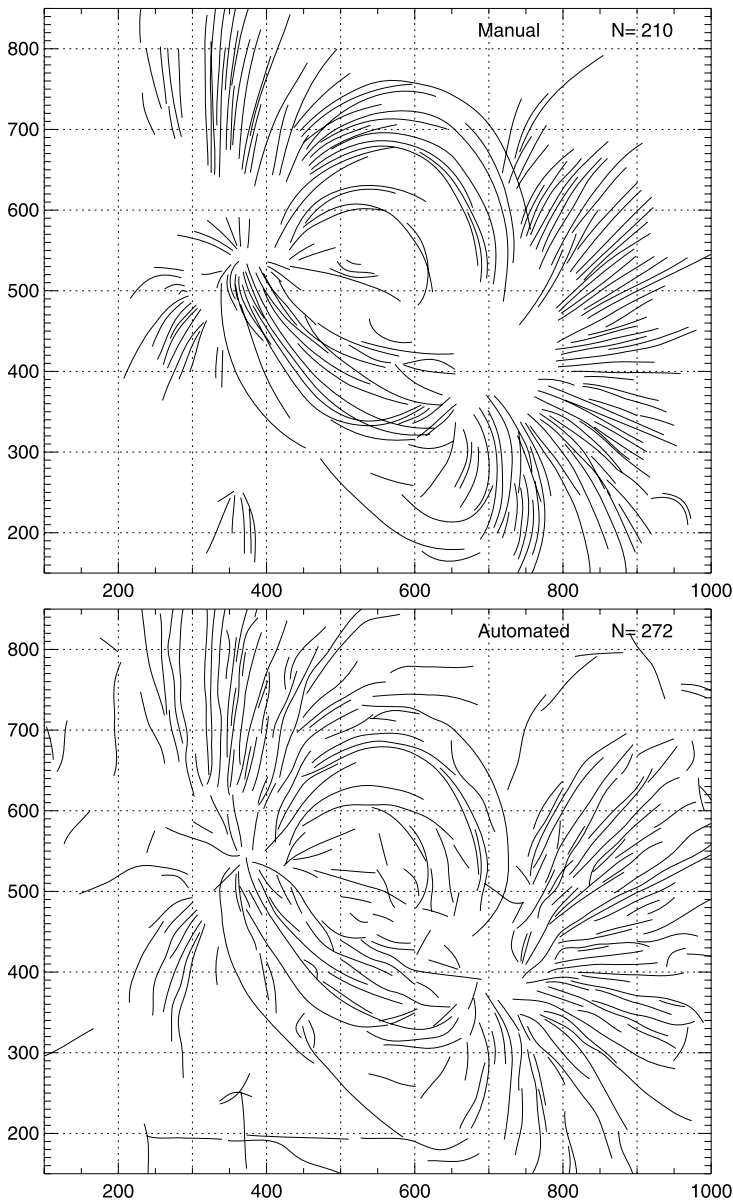
The code described here was run with this input image with the following control parameters:  $L_{\text{box}} = 30$  pixels,  $w = 3$  pixels,  $n_{\text{sm}} = 9$  pixels, and a threshold of  $N_\sigma = 1$ ,  $q_{\text{fill}, \text{min}} = 0.25$ ,  $\Delta\varphi = 30^\circ$ , and a minimum loop length of  $L_{\text{min}} = 30$ .

The computation time of this run amounted to 5.7 seconds (on a Mac OS X, Version 10.5.6, with  $2 \times 3.2$  GHz Quad-Core Intel Xeon processor and 32 GB 800 MHz DDR2 FB-DIMM memory).

#### 3.1. Comparison of Manual and Automated Tracings

The manual tracings of 210 loop structures are shown in Figure 7 (top panel), which is identical to the manual tracings shown in Aschwanden *et al.* (2008a). The results of automated tracing with the new code are shown in Figure 7 (bottom panel), amounting to  $N_{\text{loop}} = 272$  loop structures with lengths of  $L \geq L_{\text{min}} = 30$  pixels. The two tracings show a significantly better agreement than the results of five previous codes (OCM, DAM, UDM, ODM, RAS) shown in Figures 3 to 5 in Aschwanden *et al.* (2008a). The previous codes show fewer and shorter loop segments than detected manually, while the new OCCULT code detects a similar number of loops as well as similar lengths of the detected loop segments. Thus, the new code seems to perform better in terms of detection efficiency and completeness.

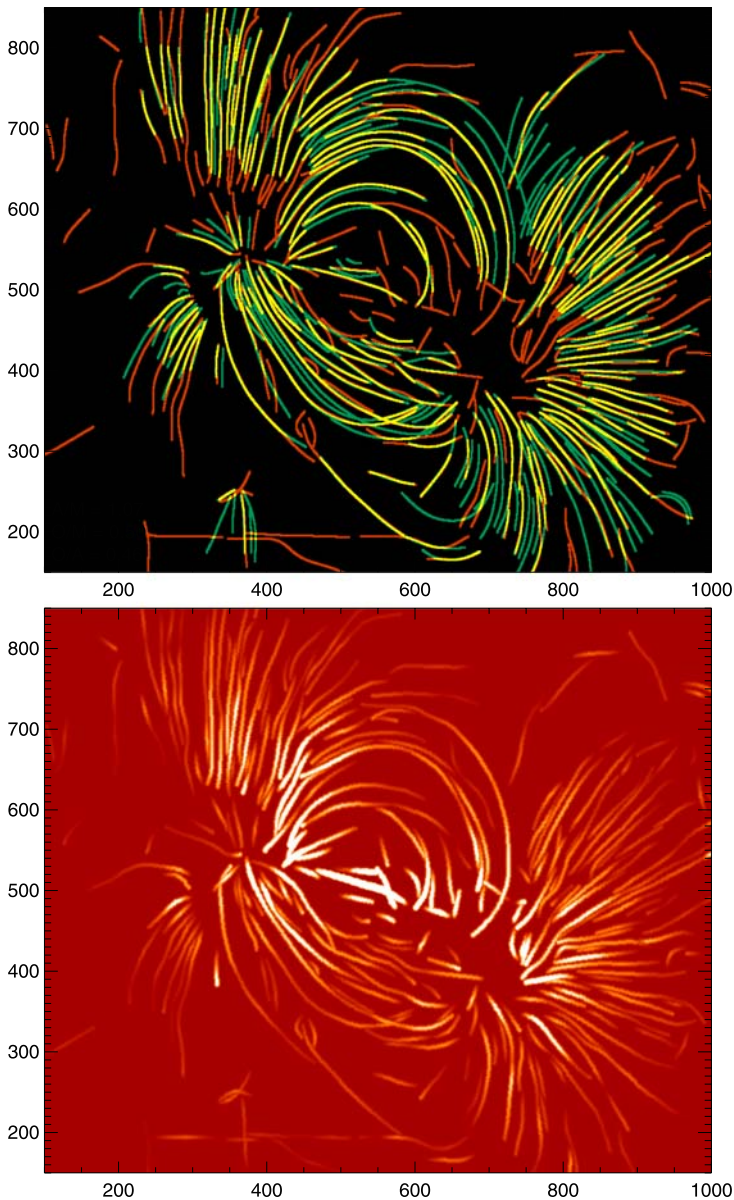
The remaining differences between manual and automated tracings need a careful assessment. It is not *a priori* clear whether the manual tracing method yields the best results,



**Figure 7** Comparison of manual (top) with automated loop tracings (bottom) in the pixel ranges of  $x = 200, \dots, 1000$  and  $y = 150, \dots, 850$  of the original image shown in Figure 2.

because visual perception often interpolates patterns that are not detectable (such as the famous Martian canals claimed by Giovanni Schiaparelli in 1877). Hence, it is interesting to inspect the differences between visual and automated detection in detail. For this purpose we created a “congruency map”, which is an overlay of the automated tracings (red curves in Figure 8, top panel) onto the manual tracings (green curves in Figure 8, top panel), with the overlapping loop segments (within  $\pm 1$  pixel) marked with a different color (yellow). First





**Figure 8** Top: A “congruency map” that allows detailed comparisons of manual (green and yellow) and automated loop tracings (red and yellow), with the overlapping pixels colored in yellow. Bottom: A “curvi-linear map” is rendered by convolving the traced loop segments with a 2D Gaussian kernel function and a flux intensity scaled to the highpass-filtered image. This presents the identical field-of-view as Figures 2 and 7 with identical pixel numeration on the axes.

of all we find a similar number of traced pixels, with a ratio of  $N_{\text{auto}}/N_{\text{manu}} = 1.07$ . The overlapping pixels constitute a fraction of 50% of the manually traced loops, and a fraction of 46% of the automatically traced loop pixels. These loop segments with overlapping pixels



are the most reliable detections, because they were detected with two independent methods. Among the features detected by the automated code only (red curves in Figure 8 top), we notice extensions of manually detected loop segments (which are probably reliable detections too), short segments in the center of the active region that may represent elongated chains of moss features (compare with Figure 2), as well as a few instrumental artifacts (horizontal streaks in the bottom left corner of image). Overall we estimate about 80%–90% of the automated detection features to be parts of coronal loops. On the other hand, features detected manually only (green curves in Figure 8 top) are generally aligned with adjacent loops, and thus they might be not real but interpolated by visual perception, or they may not have been properly resolved with the automated code (since we used a finite loop half width ( $w$ ) of three pixels).

For a better visualization of the detected loops we produce a so-called “curvi-linear map”, by convolving the automatically-detected loops pixelwise with a 2D Gaussian kernel function, with a peak amplitude corresponding to the flux value in the highpass-filtered image and a Gaussian width of 1 pixel. The result is shown in Figure 8 (bottom panel), which can be compared with the original or highpass-filtered image (Figure 2). The curvi-linear map shows only elongated structures, such as coronal loops or filaments, but efficiently filters out irregular textures (such as the reticulated moss features) or data noise. It is essentially a noise-free loop map. It may be a useful tool to display coronal loops in a noise-free rendering without dynamic range and confusion problems.

### 3.2. Loop Size Distribution

In order to compare different automated-detection codes in a quantitative way, we introduced the following quantities in Aschwanden *et al.* (2008a): the cumulative loop-length distribution  $N(>L)$ , the maximum detected loop length  $L_{\max}$ , the median loop length  $L_{\text{med}}(L > L_{\min})$  of loops above a lower limit of  $L_{\min}$ , and the ratio of detected loops compared with visually detected loops  $N_{\text{auto}}/N_{\text{manu}}$  (see Tables 3 and 4).

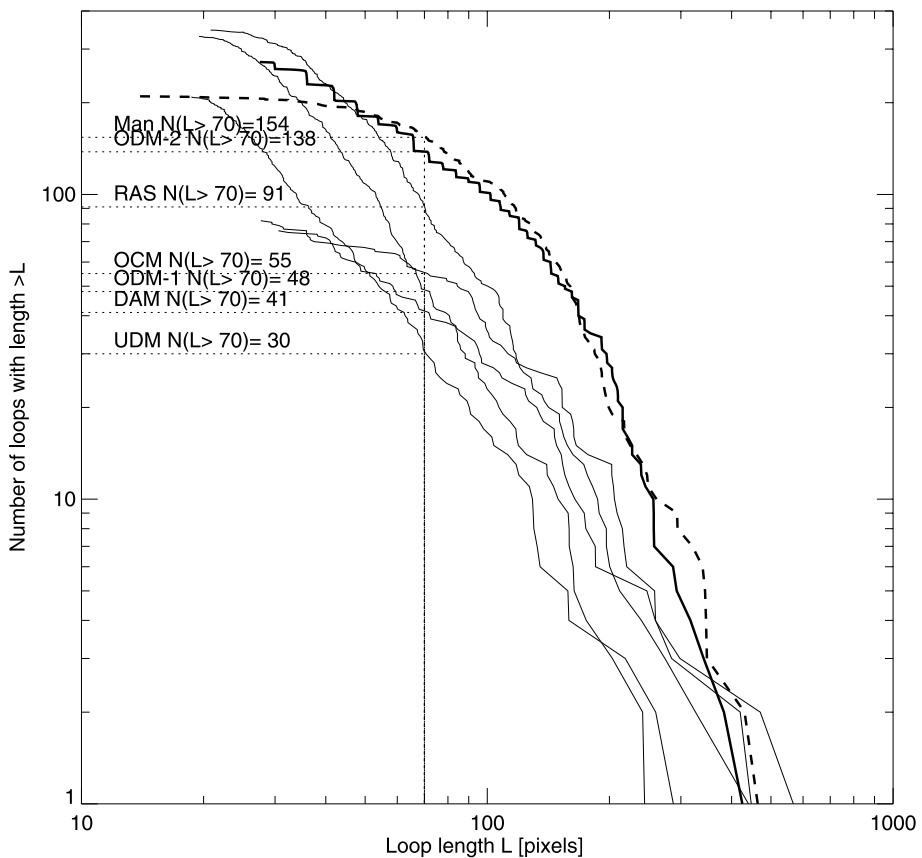
We show the cumulative loop-length distribution  $N(>L)$  in Figure 9, for the visual/manual detection, the new numerical OCCULT code, as well as the previously compared five codes shown in Figure 6 of Aschwanden *et al.* (2008a). The diagram clearly demonstrates that the new code OCCULT most closely matches the length distribution of visually traced loops, which also implies that the new code statistically detects longer loops [vertical offset in histogram  $N(>L)$ ] and has a higher detection efficiency [horizontal offset in histogram  $N(>L)$ ]. The number of visually-detected loops with a length of  $L > 70$  amounts to  $N_0 = 154$  for visual detection,  $N = 138$  for the new code OCCULT, and is in the range of  $N = 30 - 91$  for the older codes. Hence, the new code detects a relative fraction of  $N/N_0 \approx 90\%$ , while the older codes have  $N/N_0 = 20\% - 60\%$ , compared with visual detection.

**Table 3** Loop parameters measured by manual/visual detection.

Observation date time (UT)	Number of detected loops ( $N_0$ )	Maximum loop length ( $L_0$ )	Median loop length ( $M_0$ )
19 May 1998, 22:21	193	463	112
25 Aug 1998, 04:00	129	302	67
06 Nov 1999, 02:22	90	774	140
14 Jul 2000, 16:49	85	276	83
09 Nov 2000, 02:21	92	455	99

**Table 4** Ratio of automatically to manually detected loop parameters.

Observation date time (UT)	Threshold significance level [ $N_\sigma$ ]	Minimum loop length [ $L_{\min}$ ]	Ratio of detected loops [ $N/N_0$ ]	Ratio of maximum length [ $L/L_0$ ]	Ratio of median length [ $M/M_0$ ]	Comp. time [s]
19 May 1998, 22:21	1.0	45	1.05	0.92	0.91	5.7
25 Aug 1998, 04:00	1.5	30	1.03	0.95	0.97	3.5
06 Nov 1999, 02:22	1.0	50	1.10	1.03	0.64	5.9
14 Jul 2000, 16:49	2.0	50	1.18	0.93	0.93	5.9
09 Nov 2000, 02:21	2.0	50	1.13	0.74	0.90	2.6



**Figure 9** The cumulative size distribution [ $N(>L)$ ] of detected loop lengths ( $L$ ) is shown for manual tracing (dashed curve), the new OCCULT code (thick curve), and the five previously compared codes (RAS, OCM, ODM, DAM, and UDM; thin curves), with the number of loops  $N(>70$  pixels) listed in the legend. Note that the new OCCULT code exhibits a superior performance and matches the distribution of manual tracing closest.

Comparing the maximum detected loop length, we find a longest loop of  $L = 463$  pixels for manual tracing,  $L = 425$  pixels for the new code OCCULT, which represents 92%, while the older five codes fall in the range of 53% to 122%. The longest loop reflects the ability of a code to continuously trace a loop structure without being side-tracked, but this measure is somewhat approximate, because it quantifies the behavior for one (*i.e.*, the longest) loop only.

A statistically more robust measure is the median length ( $L_{\text{med}}$ ) of detected loops, above some common, minimum loop-length limit. Following the previous study, we set this limit at  $L_{\text{min}} = 70$  pixels. For manual tracing we found a median length of  $L_{\text{med},0} = 130$  pixels, while we find an almost identical value of  $L_{\text{med}} = 131$  pixels with the new OCCULT code, *i.e.*,  $L_{\text{med}}/L_{\text{med},0} = 1.01$ ; the older five codes have values in the range of  $L_{\text{med}}/L_{\text{med},0} = 0.31 - 0.93$  (see Table 1 in Aschwanden *et al.*, 2008a).

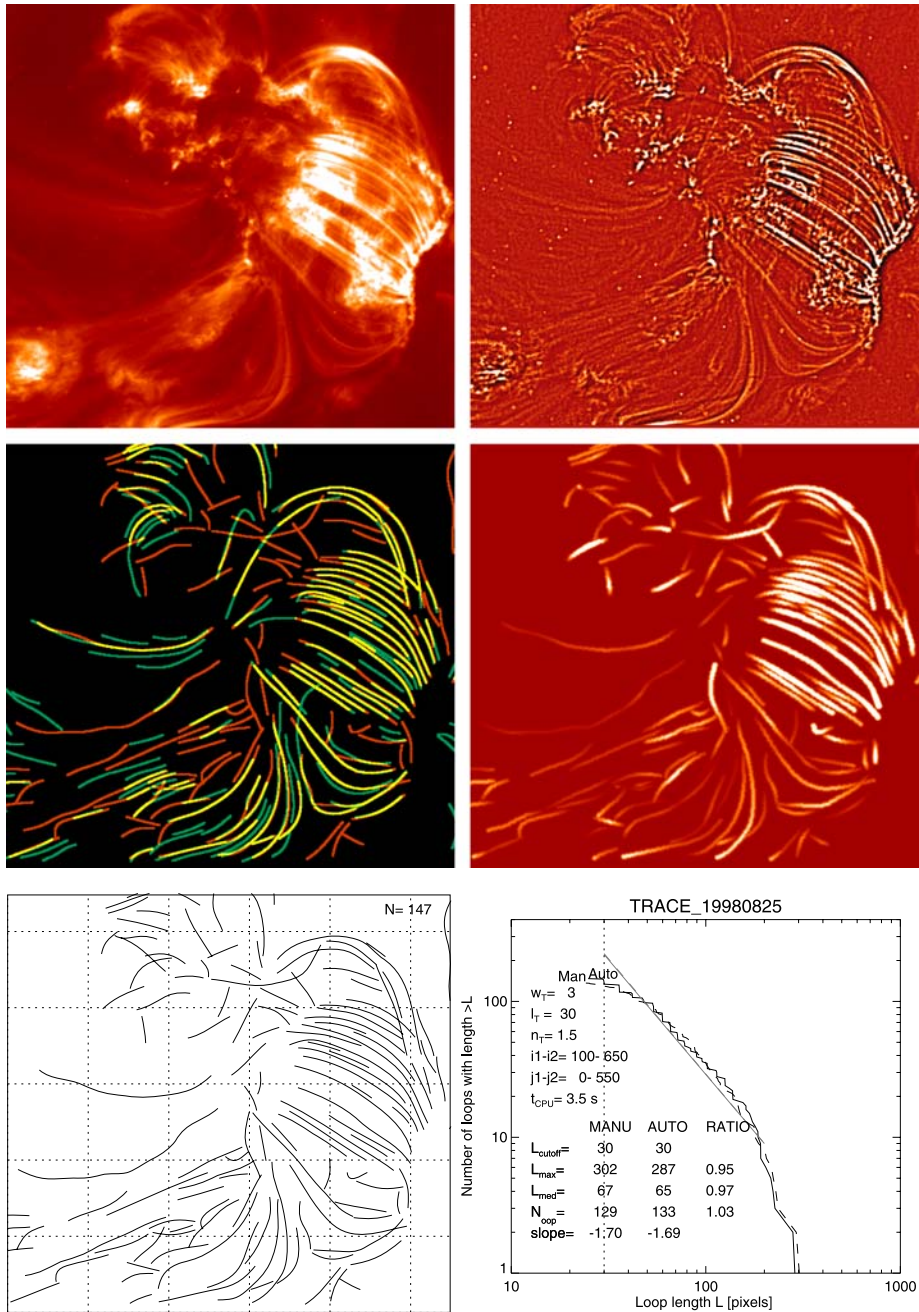
These parameters all characterize the cumulative length distribution in slightly different ways, but coincide between different detection codes if the cumulative length distributions are near-identical. Figure 9 clearly demonstrates that the visual and automated OCCULT detection produce almost identical distributions, and thus it is no surprise that the powerlaw slopes also are nearly identical ( $\alpha_{\text{manu}} = -2.79 \pm 0.04$  vs.  $\alpha_{\text{auto}} = -2.78 \pm 0.07$ ). These quantitative numbers establish that the OCCULT code is superior to the five old numerical codes and can be considered as the first numerical algorithm of automated loop detection to break even with visual detection.

#### 4. Applications to Observations

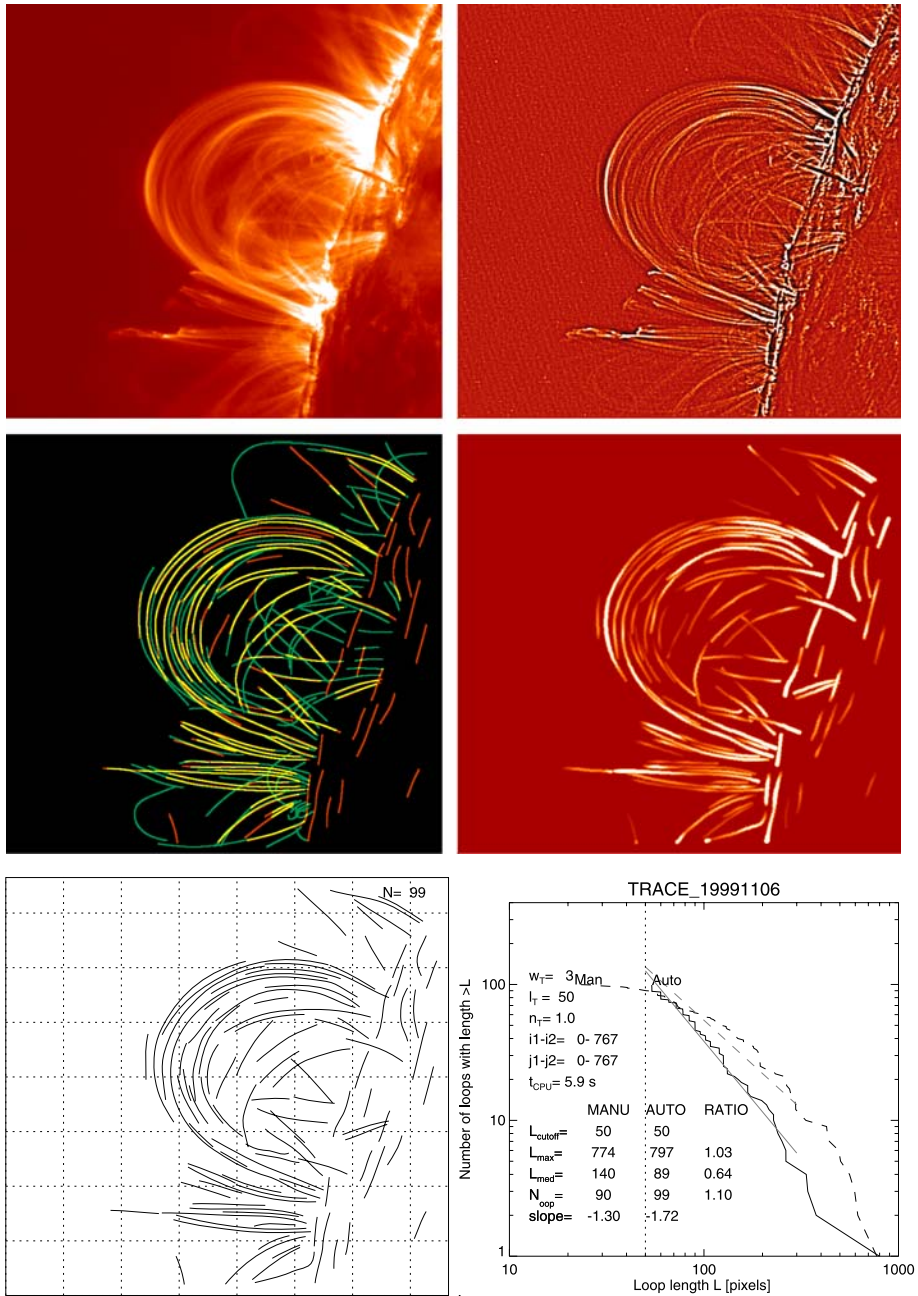
In this section we apply the new automated loop-detection code to four additional images, which are presented in Figures 10 to 13, and the results are summarized in Tables 3 and 4. The additional observations were all selected for their richness of loop structures and different geometric perspectives, and most of them appeared in previous TRACE data-analysis publications.

The image of 25 August 1998, shown in Figure 10, has been used for testing the first pioneering automated loop-detection algorithms by Strous (2000), using a pixel-labeling algorithm that detects brightness gradients between next neighbors. The TRACE 171 Å image exhibits a postflare arcade with numerous loops that all overlay a reticulated moss region, due to the special perspective near the center of the solar disk. We subject this image to manual/visual tracing and detect  $N_0 = 129$  loop structures with a maximum length of  $L_0 = 302$  pixels and a median length of  $M_0 = 67$  pixels. Running the automated code with a threshold level of  $N_\sigma = 1.5$  and a minimum length requirement of  $L_{\text{min}} = 30$  pixels, we detect a similar number of loops,  $N = 133$  (*i.e.*,  $N/N_0 = 1.03$ ), maximum loop length,  $L = 287$  pixels (*i.e.*,  $L/L_0 = 0.95$ ), and median loop length,  $M = 65$  (*i.e.*,  $M/M_0 = 0.97$ ). The cumulative loop-length distributions are almost identical (Figure 10, bottom right). The congruency map (Figure 10, middle left) shows that virtually all bright loops in the postflare arcade agree within one pixel for visual and automated detection. Therefore, this run can be considered as a successful automated detection that closely matches visual perception.

The image of 6 November 1999 (Figure 11) has been featured in numerous scientific and popular publications because of the discovery of fundamental physical properties that are relevant to coronal heating, such as the super-hydrostatic density scale heights, the near-isothermality, and the overpressure of coronal loops, which yield strong evidence for a spatially nonuniform, footpoint-driven heating process (Aschwanden, Nightingale, and Alexander, 2000; Aschwanden, Schrijver, and Alexander, 2001). We subject this image to manual

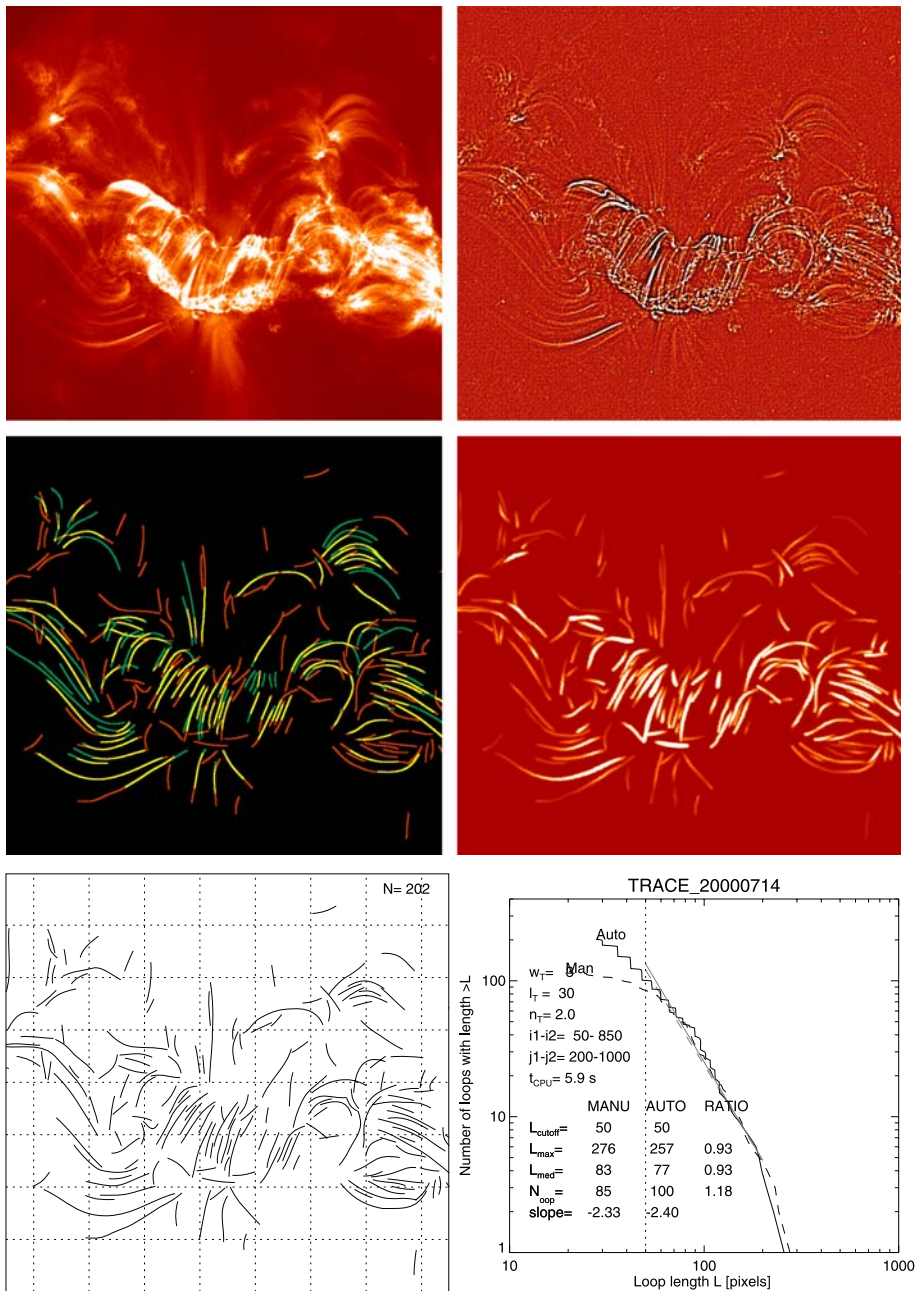


**Figure 10** Loop tracing of TRACE 171 Å 25 August 1998, 04:00:07 UT image, displaying the pixel range of  $x = 100, \dots, 675$ ,  $y = 0, \dots, 550$  of the original image spatial range. Representation similar to Figures 2, 7, 8, and 9.

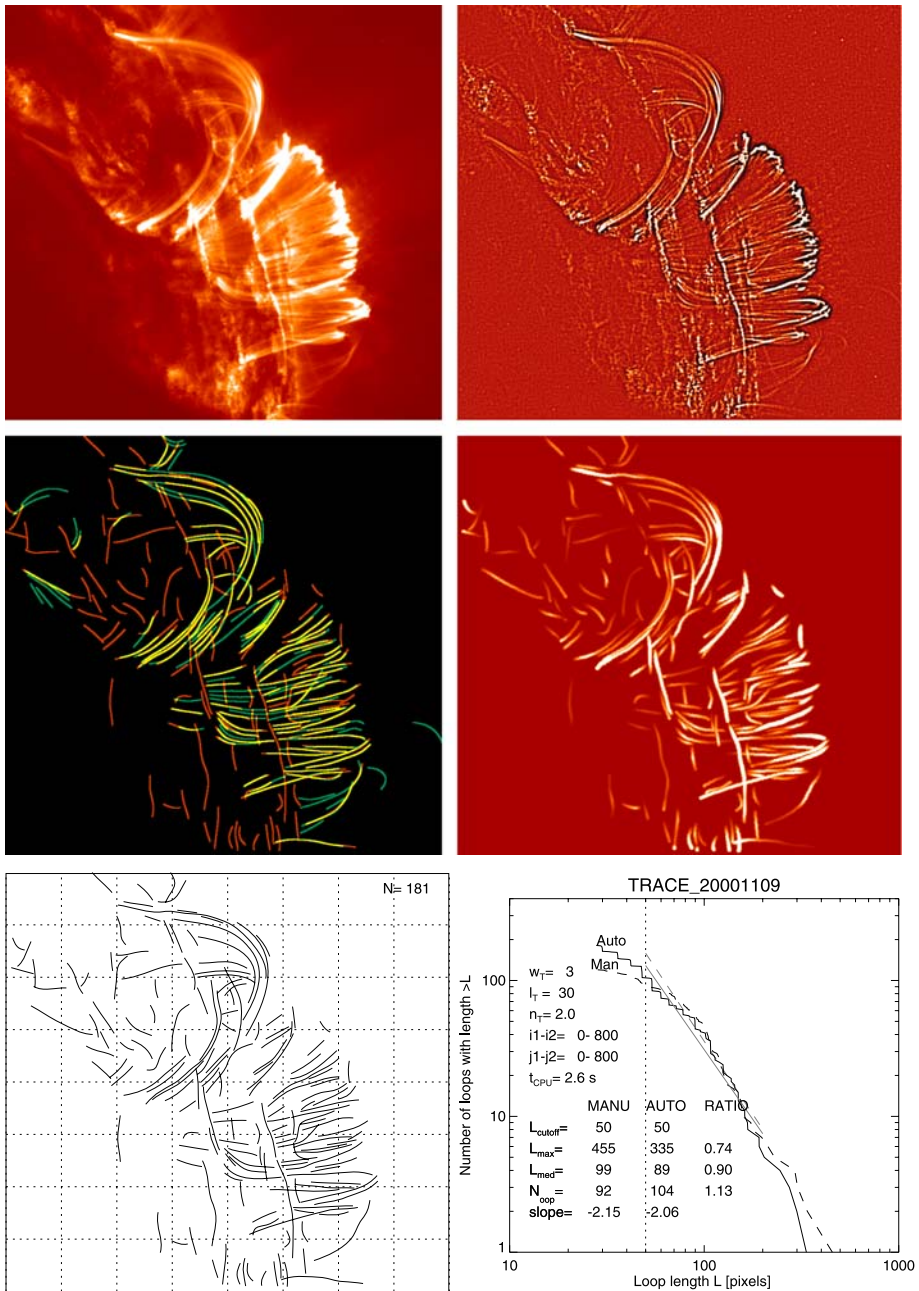


**Figure 11** Loop tracing of TRACE 171 Å 06 November 1999, 02:22:06 UT image, displaying the full size of the original  $768 \times 768$  image spatial range.





**Figure 12** Loop tracing of TRACE 171 Å 14 July 2000, 16:49 UT image, displaying the pixel range of  $x = 50, \dots, 850$ ,  $y = 200, \dots, 1000$  of the original image spatial range.



**Figure 13** Loop tracing of TRACE 171 Å 09 November 2000, 02:21:31 UT image, displaying the pixel range of  $x = 0, \dots, 800$ ,  $y = 0, \dots, 800$  of the original image spatial range.



tracing and detect  $N_0 = 90$  loops, a maximum length of  $L = 774$  pixels, and a median length of  $M = 140$  pixels, while the automated detection with a threshold of  $N_\sigma = 1.0$  and a minimum length limit of  $L_{\min} = 50$  yields a detection efficiency of  $N/N_0 = 1.10$ , and loop-length ratios of  $L/L_0 = 1.03$  and  $L/L_0 = 0.64$ . Because this image shows a particular perspective right above the solar limb, projection effects of overlaying loops are more severe than for a top-down perspective near disk center, such as in the first (Figure 8) and second case (Figure 10), which might explain the shorter median length of automatically-detected loop segments, because the loops are aligned along the line-of-sight and thus are difficult to separate and more likely to be broken up into smaller segments.

The image of 14 July 2000 (Figure 12) is also known as the “Bastille-Day flare” and the double-ribbon arcade has been dubbed a “slinky” feature. The geometric, magnetic, hydrodynamic structure, and geomagnetic impact of this flare have been studied in a large number of publications (e.g., Aulanier *et al.*, 2000; Andrews, 2001; Aschwanden and Alexander, 2001; Somov *et al.*, 2002; Tian, Wang, and Wu, 2002; Yan and Huang, 2003; Tsiklauri *et al.*, 2004; Wang *et al.*, 2005). Subjecting this image to manual/visual loop detection we traced  $N_0 = 85$  loop structures and a similar number ( $N = 100$ , i.e.,  $N/N_0 = 1.18$ ) with automated detection. Also the cumulative distribution of loop lengths (Figure 12, bottom right) and the maximum and median loop lengths are almost identical in visual and automated detection (Table 4). However, the detection of postflare loops is challenging because of the strong moss background for a disk-center view, which probably explains why the traced loops appear to be sporadically distributed and do not coherently render the cylindrical geometry of the full postflare arcade.

The image of 9 November 2000 (Figure 13) also shows a rich postflare arcade with numerous loops, with part of the arcade axis aligned along the solar limb. We manually traced  $N_0 = 92$  loops and retrieved a similar number with automated detection ( $N = 104$ , i.e.,  $N/N_0 = 1.13$ ), using a threshold level of  $N_\sigma = 2.0$  and a minimum length limit of  $L_{\min} = 50$ . Also the maximum loop length ( $L/L_0 = 0.74$ ), median length ( $M/M_0 = 0.90$ ), and the cumulative loop-length distribution (Figure 13, bottom right) are similar. The traced loops above the limb have a remarkable fidelity and congruency with visual tracings (Figure 13, middle left), but a number of curvi-linear features appear inside the limb that are more or less aligned with the limb. These features are most likely not loop structures, but rather curvi-linear chains of moss structures that appear straight near the limb due to the strong foreshortening. A similar effect is also seen for the 6 November 1999 image (Figure 11, middle left). In future refinements of automated loop tracing codes, foreshortening effects near the solar limb should be taken into account in the definition of feature extraction.

## 5. Discussion and Conclusions

The development of automated detection codes is a new subfield of solar research that became a necessity due to the rapidly increasing amount of imaging data as well as due to the quest for objectivity and precision in data analysis. There already exist a number of automated feature-recognition tools that are specialized for each solar phenomenon, such as active regions, sunspots, filaments, flares, *etc.* (see the recent review by Aschwanden (2010) and references therein). It is clear that no automated feature-recognition code is universal enough to detect all types of features in an optimum way, hence each feature requires a customized code. The phenomenon of coronal loops has its own intrinsic properties, such as large curvature radii (compared with their widths) due to the small-scale heating processes that drive chromospheric upflows into large-scale coronal magnetic flux tubes. Thus, it

makes sense to use this property of large curvature radii to enable more efficient detection of coronal loops, a principle that we used in this newly developed numerical code here. It turns out that the new code produces results that are close to visual perception, regarding fidelity, congruency, and detection efficiency, as quantified by the cumulative distribution of detected loop lengths, which can be considered as a first break-even with visual perception. The code is also relatively fast, completing the automated detection of a  $1024 \times 1024$  pixel image within a few seconds on a state-of-the-art computer (Table 4). While the new code represents a significantly advanced tool compared with previous detection codes, there are still some limitations that could be improved in future developments. One source of confusion is foreshortened moss regions near the limb, which have curvature radii corresponding to the solar radius and alignment with the limb, which could be taken into account based on the heliographic position. Another source of confusion is closely-spaced, near-parallel loops, which are near co-spatial and have almost identical curvature radii, which can only be properly traced by simultaneously tracking them both. Also, our code is optimized for the detection of loops with a characteristic width, but is more prone to fail when loops with a wide range of widths are present, for which a multi-scale method could be more efficient in future.

The new automated loop-detection code should be useful for modeling of the coronal magnetic field (see, e.g. Gary and Alexander, 1999; Schrijver *et al.*, 2005, 2006, 2008; DeRosa *et al.*, 2009; Sandman *et al.*, 2009), for automated stereoscopic triangulation of coronal loops (see, e.g. Aschwanden *et al.*, 2008b), for measurements of cross-sectional and width variations of loops (see, e.g., Klimchuk, 2000; Lopez Fuentes, Demoulin, and Klimchuk, 2008), for hydrodynamic modeling (see, e.g. Winebarger and Warren, 2005; Warren and Winebarger, 2007), modeling of coronal heating (see, e.g., Aschwanden, Schrijver, and Alexander, 2001; Schrijver *et al.*, 2004; A. Van Ballegooijen, private communication, 2009), rendering of active regions and full-Sun visualizations (see, e.g. Schrijver *et al.*, 2004; Lundquist *et al.*, 2008; Warren and Winebarger, 2007), or tracking of MHD (kink-mode) loop oscillations (see, e.g. Aschwanden *et al.*, 1999b; Nakariakov *et al.*, 1999), *etc.* There are numerous potential applications of an automated loop-detection code in almost the entire field of coronal physics (see, e.g. Aschwanden, 2004). Alternative applications of this new code to other curvi-linear features are beyond the scope of this paper, but obvious candidates are coronal quiescent and eruptive filaments, streamers, plumes, jets, CME features (leading edges, flanks, flux ropes), chromospheric features (fibrils, spicules, waves), or photospheric features (sunspot penumbral filaments). However, we expect that this code will do best for coronal loops, but needs further customization for other solar phenomena.

**Acknowledgements** I acknowledge helpful suggestions of the referee and discussions with Karel Schrijver, Mark DeRosa, Allen Gary, Jake Lee, Bernd Inhester, David Alexander, James McAteer, Peter Gallagher, Alex Young, Alex Engels, Narges Fathalian, Fatemeh Amirkhanlou, Hossein Safari, and participants of the *3rd Solar Image Processing Workshop* in Dublin, Ireland, 6 – 8 September 2006, and the *4th Solar Image Processing Workshop* in Baltimore, Maryland, 26 – 30 October 2008. Part of the work was supported by the NASA TRACE (NASS-38099) and NASA STEREO/SECCHI contracts (N00173-02-C-2035, administrated by NRL).

## References

- Andrews, M.D.: 2001, *Solar Phys.* **204**, 181.
- Aschwanden, M.J.: 2004, *Physics of the Solar Corona – An Introduction*, Praxis Publishing Ltd., Chichester, Springer, Berlin.
- Aschwanden, M.J.: 2005, *Solar Phys.* **228**, 339.

- Aschwanden, M.J.: 2010, Image processing techniques and feature recognition in solar physics. *Solar Phys.*, doi:[10.1007/s11214-009-9505-x](https://doi.org/10.1007/s11214-009-9505-x).
- Aschwanden, M.J., Alexander, D.: 2001, *Solar Phys.* **204**, 91.
- Aschwanden, M.J., Nightingale, R.W., Alexander, D.: 2000, *Astrophys. J.* **541**, 1059.
- Aschwanden, M.J., Schrijver, C.J., Alexander, D.: 2001, *Astrophys. J.* **550**, 1036.
- Aschwanden, M.J., Newmark, J.S., Delaboudinière, J.-P., Neupert, W.M., Klimchuk, J.A., Gary, G.A., Portier-Fozzani, F., Zucker, A.: 1999a, *Astrophys. J.* **515**, 842.
- Aschwanden, M.J., Fletcher, L., Schrijver, C., Alexander, D.: 1999b, *Astrophys. J.* **520**, 880.
- Aschwanden, M.J., Lee, J.K., Gary, G.A., Smith, M., Inhester, B.: 2008a, *Solar Phys.* **248**, 359.
- Aschwanden, M.J., Wülser, J.P., Nitta, N., Lemen, J.: 2008b, *Astrophys. J.* **679**, 827.
- Aulanier, G., DeLuca, E.E., Antiochos, S.K., McMullen, R.A., Golub, L.: 2000, *Astrophys. J.* **540**, 1126.
- Berger, T.E., De Pontieu, B., Schrijver, C.J., Title, A.M.: 1999, *Solar Phys.* **190**, 409.
- DeRosa, M.L., Schrijver, C.J., Barnes, G., Leka, K.D., Lites, B.W., Aschwanden, M.J., Amari, T., Canou, A., McTiernan, J.M., Regnier, S., Thalmann, J., Valori, G., Wheatland, M.S., Wiegelmann, T., Cheung, M.C.M., Conlon, P.A., Fuhrmann, M., Inhester, B., Tadesse, T.: 2009, *Astrophys. J.* **696**, 1780.
- Durak, N., Nasraoui, O., Schmelz, J.: 2009, *Pattern Recogn.* **42**, 2481.
- Durak, N., Nasraoui, O., Schmelz, J.: 2010, *Solar Phys.*, submitted.
- Freeland, S.L., Handy, B.N.: 1998, *Solar Phys.* **182**, 497.
- Gary, A., Alexander, D.: 1999, *Solar Phys.* **186**, 123.
- Gonzalez, R.C., Woods, R.E.: 2008, *Digital Image Processing*, 3rd edn., Pearson Prentice Hall, Upper Saddle River.
- Handy, B.N., Acton, L.W., Kankelborg, C.C., Wolfson, C.J., Akin, D.J., Bruner, M.E., Carvalho, R., Catura, R.C., Chevalier, R., Duncan, D.W., et al.: 1999, *Solar Phys.* **187**, 229.
- Inhester, B., Feng, L., Wiegelmann, T.: 2007, *Solar Phys.* **248**, 379.
- Jähne, B.: 2005, *Digital Image Processing*, 6th edn., Springer, Berlin.
- Klimchuk, J.A.: 2000, *Solar Phys.* **193**, 53.
- Lee, J.K., Newman, T.S., Gary, G.A.: 2004, In: Kittler, J., Petrou, M., Nixon, M.S. (eds.) *Proc. 17th International Conference on Pattern Recognition (ICPR)*, IEEE-CS, Washington, 315. doi:[10.1109/ICPR.2004.1333766](https://doi.org/10.1109/ICPR.2004.1333766).
- Lee, J.K., Newman, T.S., Gary, G.A.: 2006a, *Pattern Recogn.* **39**, 246.
- Lee, J.K., Newman, T.S., Gary, G.A.: 2006b, In: *Proc. 7th IEEE SSIAI*, Denver, CO, 91.
- Lopez Fuentes, M.C., Demoulin, P., Klimchuk, J.A.: 2008, *Astrophys. J.* **673**, 586.
- Lundquist, L.L., Fisher, G.M., Metcalf, T.R., Leka, K.D., McTiernan, J.M.: 2008, *Astrophys. J.* **689**, 1388.
- McAteer, R.T., Kestener, P., Arneodo, A., Khalil, A.: 2010, *Solar Phys.*, submitted.
- Nakariakov, V.M., Ofman, L., DeLuca, E., Roberts, B., Davila, J.M.: 1999, *Science* **285**, 862.
- Pao, Y.-H.: 1996, *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley, New York.
- Raghupathy, K.: 2004, PhD Thesis, Cornell University.
- Ripley, B.D.: 1996, *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge.
- Sandman, A., Aschwanden, M.J., DeRosa, M., Wülser, J.P., Alexander, D.: 2009, *Solar Phys.* **259**, 1.
- Schrijver, C.J., Sandman, A.W., Aschwanden, M.J., DeRosa, M.L.: 2004, *Astrophys. J.* **615**, 512.
- Schrijver, C.J., DeRosa, M.L., Title, A.M., Metcalf, T.R.: 2005, *Astrophys. J.* **628**, 501.
- Schrijver, C.J., DeRosa, M., Metcalf, T.R., Liu, Y., McTiernan, J., Regnier, S., Valori, G., Wheatland, M.S., Wiegelmann, T.: 2006, *Solar Phys.* **235**, 161.
- Schrijver, C.J., DeRosa, M.L., Metcalf, T., Barnes, G., Lites, B., Tarbell, T., McTiernan, J., Valori, G., Wiegelmann, T., Wheatland, M.S., Amari, T., Aulanier, G., Demoulin, P., Fuhrmann, M., Kusano, K., Regnier, S., Thalmann, J.K.: 2008, *Astrophys. J.* **675**, 1637.
- Sellah, S., Nasraoui, O.: 2008, In: *Tools with Artificial Intelligence, ICTAI'08, 20th IEEE Internat. Conf.* **2**, 45. doi:[10.1109/ICTAI.2008.147](https://doi.org/10.1109/ICTAI.2008.147).
- Smith, M.: 2005, <http://www.mssl.ucl.ac.uk/twiki/bin/view/SDO/LoopRecognition>.
- Somov, B.V., Kosugi, T., Hudson, H.S., Sakao, T.: 2002, *Astrophys. J.* **579**, 863.
- Steger, L.H.: 1996, Technical Report FGBV-96-03, Forschungsgruppe Bildverstehen (FG BV), Informatik IX, Technische Universität München, Germany.
- Strous, L.H.: 2000, [http://www.lmsal.com/aschwand/stereo/2000\\_easton/cdaw.html](http://www.lmsal.com/aschwand/stereo/2000_easton/cdaw.html).
- Tian, L., Wang, J., Wu, D.: 2002, *Solar Phys.* **209**, 375.
- Tsiklauri, D., Aschwanden, M.J., Nakariakov, V.M., Arber, T.D.: 2004, *Astron. Astrophys.* **419**, 1149.
- Wang, H., Liu, C., Deng, Y., Zhang, H.: 2005, *Astrophys. J.* **627**, 1031.
- Warren, H.P., Winebarger, A.R.: 2007, *Astrophys. J.* **666**, 1245.
- Winebarger, A.R., Warren, H.P.: 2005, *Astrophys. J.* **626**, 543.
- Yan, Y., Huang, G.: 2003, *Space Sci. Rev.* **107**, 111.