QUADRATIC MODELS FOR CURVED LINE DETECTION IN SAR CCD

Davis E. King and Rhonda D. Phillips

MIT Lincoln Laboratory
{davis.king,rhonda.phillips}@ll.mit.edu

ABSTRACT

Synthetic Aperture Radar Coherent Change Detection (SAR CCD) is a sensitive change detector capable of finding ground surface height changes on the order of a radar wavelength. SAR CCD also has a high false alarm rate, with many false alarms caused by vegetation that is easily displaced by wind, rain, etc. This work introduces a robust method for detecting long curving structures in this challenging medium. To demonstrate the accuracy of the method, we test it on the task of automatically distinguishing curving tire tracks from background clutter.

Index Terms— SAR, coherent change detection, image filtering

1. INTRODUCTION

Synthetic Aperture Radar Coherent Change Detection (SAR CCD) is a sensitive change detector that reveals scene changes that are on the order of radar wavelength rather than range resolution. SAR CCD has applications in activity monitoring (by detecting track imprints) [1] [2], search and rescue operations [3], and more. The detection of vehicle tracks is possible because vehicles make small changes on the ground that slightly alter SAR magnitude and phase values. The automatic detection of these curvilinear tracks is difficult because the coherence values on such a track are not uniformly low. Furthermore, there are many naturally occurring false alarms (low coherence that is not the result of scene change) throughout the image that confuse line detectors. In this paper, we introduce a curvilinear line detection method designed specifically for SAR CCD images. The method uses two steps, one to detect curve points, and one to join points together to form curves. The first step accounts for the nonuniform coherence values on line segments, and the second step is necessary to eliminate false alarms. The following section describes SAR CCD processing and elaborates on the difficulty in detecting line segments in SAR CCD. Section 3 describes the curve detection algorithm, and Section 4 presents experimental results.

2. SAR CCD

SAR CCD is formed using two registered SAR images collected from identical geometries. The collection geometry requirement is necessary because SAR CCD exploits differences in phase to detect subtle disturbances on the ground. [4] and [5] provide a comprehensive overview of the processing required to perform coherent change detection. CCD images are coherence maps between two SAR images, f and g, where the coherence value at each pixel, γ, represents the coherence between that complex pixel in f and g. CCD is actually an estimated coherence, $\hat{\gamma}$, which is a random variable that depends on the true coherence and the number of samples used to compute the estimate [6]. The random nature of $\hat{\gamma}$ and of the noise present in f

and g contribute to different estimated coherence values along a scene change such as a track. This important property must be taken into account in any automated detector.

In addition to natural variations in coherence estimates of scene changes, there are several sources of low coherence that are not caused by scene changes. Issues with registration and phase alignment cause low coherence [4]. In areas of low radar return such as radar shadows that are predominantly noise, there is low estimated coherence. Additionally, vegetation can appear to have changed between SAR passes because leaves or grass are naturally displaced due to environmental factors. Any automatic scene change detector will have to account for these false alarms or areas of low coherence that do not indicate scene change.

Finally, while naturally occurring low coherence causes many false curve point detections, most of these curve points cannot be connected together to form valid curves. Therefore, where possible, our curved line detector connects individual curve points to find curved lines. Once connected, aggregate statistics from the entire curve are used to filter false alarms.

3. ALGORITHM DESCRIPTION

Our approach to curve detection has two steps. First, quadratic image models are used to detect points likely to lie on a curve. Next, each connected group of detection pixels is fit to a quadratic curve model and false alarms are filtered based the quality of fit and overall appearance of the resulting curve.

3.1. CURVE POINT DETECTION

Our goal is to detect pixels corresponding to curves of low coherence values relative to the surrounding area. Therefore, we model the region surrounding a candidate pixel as a locally quadratic surface, and we use the shape of this surface to determine if it is a curve point. In particular, curve points are characterized by a trough shaped surface where low coherence points lie at the bottom of the trough. In mathematical terms, this means that the Hessian matrix for the quadratic surface has one large eigenvalue and one zero (or small) eigenvalue. Moreover, the eigenvector associated with the largest eigenvalue indicates the direction perpendicular to the curve because it points along the direction of greatest curvature.

To compute these quantities, we fit a quadratic function $f(x, y) = w_1 x^2 + w_2 y^2 + w_3 xy + w_4$ to the area around a pixel at location $(\hat{x}, \hat{y})$ by solving the following locally weighted regression problem,

$$\min_w \sum_{i=1}^{N} e^{\frac{-(x_i^2 + y_i^2)}{2\sigma^2}} \left(f(x_i, y_i) - \mathcal{I}(\hat{x} + x_i, \hat{y} + y_i) \right)^2,$$

where σ controls the size of the interpolation region. The sum is over the N neighboring pixels that have non-trivial values of $e^{\frac{-(x_i^2 + y_i^2)}{2\sigma^2}}$ and $\mathcal{I}(x, y)$ denotes the coherence value for a pixel at point (x, y).

To obtain the solution, let A, G, and P be defined as follows:

$$A = \begin{bmatrix} x_1^2 & y_1^2 & x_1 y_1 & 1 \\ x_2^2 & y_2^2 & x_2 y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_N^2 & y_N^2 & x_N y_N & 1 \end{bmatrix} \quad G = \begin{bmatrix} e^{\frac{-(x_1^2 + y_1^2)}{4\sigma^2}} & 0 & \cdots & 0 \\ 0 & e^{\frac{-(x_2^2 + y_2^2)}{4\sigma^2}} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & e^{\frac{-(x_N^2 + y_N^2)}{4\sigma^2}} \end{bmatrix} \quad P = \begin{bmatrix} \mathcal{I}(\hat{x} + x_1, \hat{y} + y_1) \\ \mathcal{I}(\hat{x} + x_2, \hat{y} + y_2) \\ \vdots \\ \mathcal{I}(\hat{x} + x_N, \hat{y} + y_N) \end{bmatrix}$$

This can be converted into a standard least-squares problem by making the following substitutions, $\bar{A} = GA$, $\bar{P} = GP$. Then the optimal parameters are given by $w = (\bar{A}^T \bar{A})^{-1} \bar{A}^T \bar{P}$. Note that the optimal parameters are obtained by applying a set of linear filters to P. That is, each filter is given by a row of $(\bar{A}^T \bar{A})^{-1} \bar{A}^T G$ (i.e. row 1 determines w_1, row 2 determines w_2,

etc.). Moreover, these filters are separable and can thus be efficiently evaluated over an entire image, even for large interpolation regions. This is a critical detail since the noise level in SAR CCD necessitates interpolating over relatively large regions.

With the optimal w in hand, the maximum eigenvalue and associated eigenvector for the Hessian of the quadratic surface are obtained from:

$$\lambda = w_1 + w_2 + \sqrt{(w_1 - w_2)^2 + w_3^2}, \quad e_\lambda = \begin{cases} \begin{bmatrix} \frac{-w_3}{2w_1 - \lambda} \\ 1 \end{bmatrix} & 2w_1 \neq \lambda \\ \begin{bmatrix} 1 \\ 0 \end{bmatrix} & 2w_1 = \lambda \end{cases},$$

In the next step, we construct a curve point detection image where each pixel contains the maximum eigenvalue from the corresponding location in the input image. A negative λ would indicate a curve made up of high coherence pixels surrounded by a low coherence region (the opposite of what we wish to find), so all negative values are converted to zero. Finally, since e_λ determines the orientation of each curve, we use it to perform non-max suppression. That is, any detection point which is not a local maximum along the line defined by e_λ is assigned to zero. The result of this can be seen in the middle of Figure 1.

3.2. CURVE MODEL FITTING

Curve point detection produces a high number of false alarm pixels. To mitigate this, we observe that true curves of interest (e.g. tire tracks) tend to follow smoothly curving trajectories. In particular, these curves can be well modeled as quadratic curves of the form $g(x, y) = v_1 x^2 + v_2 y^2 + v_3 xy + v_4 x + v_5 y = 1$. This model captures conic sections such as parabolas, ellipses, and hyperbolas.

Therefore, we group connected segments of detection pixels and determine the quadratic curve that best interpolates them. Additionally, we compute the mean distance between each detection pixel and the interpolated curve. This distance is used to quantify the quality of the model fit. To be precise, we formulate this as a regularized orthogonal regression problem defined by the following constrained optimization (let $\hat{z}_i = [\hat{x}_i; \hat{y}_i]$ denote the locations of the detection pixels):

$$\min_{v, z_1, z_2, \dots} \gamma ||v||^2 + \sum_{i=1}^{N} ||\hat{z}_i - z_i||^2 \tag{1}$$

$$\text{such that } g(z_i) = 1, \ \forall i,$$

That is, we find the quadratic model which minimizes the mean squared distance between the quadratic curve and the data points, subject to a regularization penalty. This can be solved using the L-BFGS technique combined with a penalty method to handle the constraint [7]. It is possible that the best fit is a an extremely eccentric ellipse or hyperbola that splits the points onto two disconnected curves. While this can result in a good fit in terms of low mean distance between points and the curve, these shapes do not correspond to realistic detections in SAR CCD images. To handle this case, we regularize the regression using the L2 norm of v. In particular, we first solve the optimization with a small γ and if the solution results in a disconnected hyperbolic curve we discard the results, increase γ and solve it again. If after a fixed number of attempts we still obtain a disconnected hyperbola, we drop the quadratic terms from g and use the resulting linear model.

Finally, we model pixel segments as binary variables in a graphical model where edges encode geometric connectedness information from Equation 1. We then use standard graph-cut methods to segment true curves from false alarms [8].

4. EXPERIMENTAL RESULTS

Our curve detection procedure is demonstrated in Figure 1. The middle image shows that the curve point detection produces a large number of false alarms. However, the connected segments corresponding to false alarms are small and cannot be grouped

together due to their poor quadratic model fit. The far right image shows only those segments which fit together into statistically significant tracks according to our graphical model.
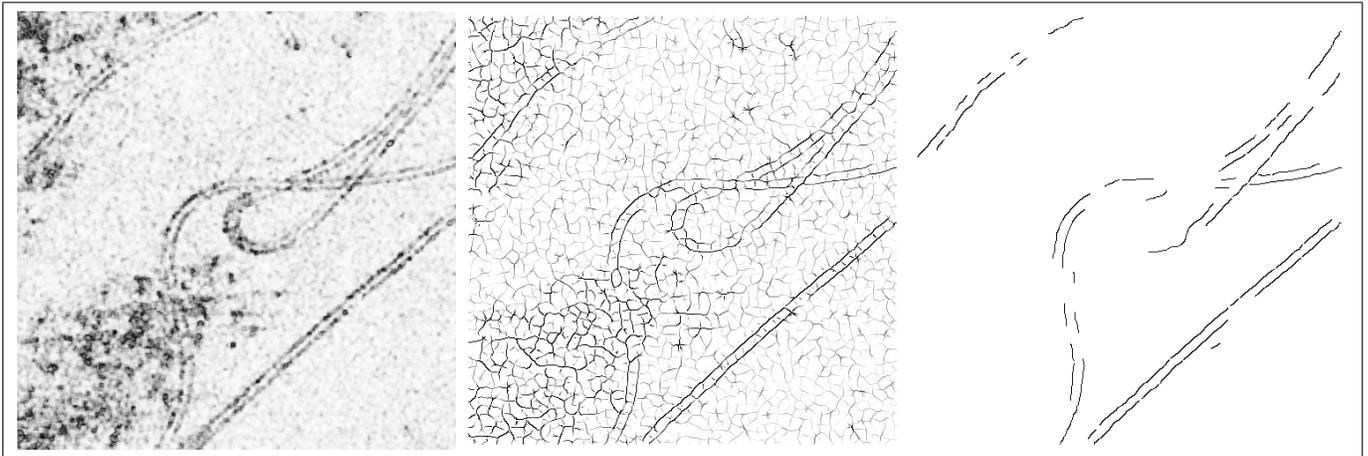

Fig. 1. From left to right: An example SAR CCD image, the output of curve point detection, and finally the output of the curve model fitting and filtering process.

5. ACKNOWLEDGEMENTS

6. REFERENCES

[1] D.G. Corr and A. Rodrigues, "Coherent change detection of vehicle movements," *IGARSS '98 - 1998 International Geoscience and Remote Sensing*, pp. 2451–2453, 1998.

[2] N. Milisavljevic, D. Closson, and I. Bloch, "Detecting potential human activities using coherent change detection," in *2nd International Conference on Image Processing Theory, Tools, and Applications (IPTA 2010)*, 2010, pp. 482–485.

[3] T.I. Lukowski and F.J. Charbonneau, "Synthetic aperture radar and search and rescue: Detection of crashed aircraft using imagery and interferometric methods," *Canadian Journal of Remote Sensing*, vol. 28, no. 6, pp. 770–781, Dec. 2002.

[4] A.W. Doerry, "SAR data collection and processing requirements for high quality coherent change detection - art. no. 694706," *Radio Sensor Technology XII*, vol. 6947, pp. 94706–94706, 2008.

[5] C.V. Jakowatz, Jr, D.E. Wahl, P.H. Eichel, D.C. Ghiglia, and P.A. Thompson, *Spotlight-mode Synthetic Aperture Radar: A Signal Processing Approach*, chapter 5, Kluwer Academic Publishers, Norwell, MA, 1996.

[6] R. Touzi, A. Lopes, J. Bruniquel, and P.W. Vachon, "Coherence estimation for SAR imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 37, no. 1, pp. 135–149, 1999.

[7] Jorge Nocedal and Stephen Wright, *Numerical Optimization, 2nd edition*, Springer, 2006.

[8] Sebastian Nowozin and Christoph H. Lampert, *Structured Prediction and Learning in Computer Vision*, vol. 6, Foundations and Trends in Computer Graphics and Vision, 2011.