Short Papers

Breakpoint Detection Using Covariance Propagation

Qiang Ji and Robert M. Haralick

Abstract—This paper presents a novel statistical approach for detecting breakpoints from chain encoded digital arcs. An arc point is declared as a breakpoint if the estimated orientations of the two fitted lines of the two arc segments immediately to the right and left of the arc point are significantly statistically different. The major contributions of this research include developing a method for analytically estimating the covariance matrix of the fitted line parameters and proposing a perturbation model to characterize the perturbation associated with each arc point.

Index Terms—Corner detection, arc segmentation, error propagation.

1 Introduction

Breakpoints are dominant points (points with sufficiently high curvatures) on the boundary of an object. They have long been important two-dimensional features for computer vision research. They have been used extensively for matching, pattern recognition, and data compression. Over the years, various algorithms have been developed for detecting breakpoints. Breakpoint detection algorithms can be roughly grouped into two categories: One is based on the detection directly from the underlying images [3], [6], [18], the other is based on digital arcs, resulting from edge detection and linking [2], [11], [8], [1], [9], [16], [17]. The research described in this paper is concerned with detecting breakpoints from digital arcs.

Breakpoint detection from digital arcs partitions a given digital arc sequence into digital arc subsequences having the property that each arc subsequence is a maximal length sequence to which it is reasonable to fit a line. Various techniques have been developed for breakpoint detection from digital arc sequences. The basis for these techniques is to identify the locations of the endpoints of each maximal line segment. Different criteria have been proposed for detecting breakpoints including maximum curvature, deflection angle, maximum deviation, and total fitting errors [1], [9], [10], [16], [8], [2], [11], [17]. A major problem with the existing approaches is that the employed criterion is not tied to a statistical analysis involving an explicit noise model, therefore making the method statistically inefficient with regard to noise. To overcome this, we have developed a statistical approach to breakpoint detection, where the breakpoint criterion is treated as a random variable and is subject to perturbation. Given an arc segment, an arc point is declared as a breakpoint if the estimated orientations of the two fitted lines of the two arc segments immediately to the right and left of the arc point are significantly statistically different. The major contributions of this research include developing a method for analytically estimating the covariance matrix of the fitted line parameters, proposing a perturbation model to characterize the perturbation associated with each arc point, and developing a hypothesis testing statistic to statistically test the difference between the estimated orientations of the two fitted lines.

This paper is arranged as follows. In Section 2, we state the problem and present the associated noise and breakpoint models. Section 3 discusses in detail the theoretical aspects of the breakpoint detector. The performance characterization and comparison of the breakpoint detector is covered in Sections 4 and 5. The paper ends in Section 6 with a discussion and conclusion of the developed approach.

2 Problem Statement

A breakpoint represents a discontinuity in the curvature of a curve. The location of the discontinuity can be approximated by the intersection of two straight lines that underlie the arc segments to the right and left of the breakpoint. Perturbation to the points on the ideal underlying lines gives rise to the observed arc segments. This section discusses the definition of the perturbation model and the breakpoint model.

2.1 Perturbation Model

Given an observed sequence of N ordered points from a line arc segment,

$$S = \left\{ (\hat{x}_n, \hat{y}_n) \middle| n = 1, \ldots, N \right\},$$

the perturbation model assumes that $(\hat{x}_n, \hat{y}_n)$ result from random perturbations to the ideal points (x_n, y_n), constrained to be on the line

$$x_n \cos\theta + y_n \sin\theta - \rho = 0, \quad n = 1, \ldots, N$$

where θ and ρ are the parameters of the underlying line that gives rise to the observed arc segment. It is further assumed that the random perturbations are independently and identically Gaussian distributed in the direction perpendicular to the underlying line. Analytically, the perturbation model can be expressed as follows

$$\begin{pmatrix} \hat{x}_n & \hat{y}_n \end{pmatrix}^t = \begin{pmatrix} x_n & y_n \end{pmatrix}^t + \xi_n \begin{pmatrix} \cos\theta & \sin\theta \end{pmatrix}^t \tag{1}$$

where ξ_n are independently and identically distributed as N(0, σ^2).

2.2 Breakpoint Model

For a piecewise linear approximation of a curve, breakpoints are the endpoints of each line segment. Thus, an endpoint is a breakpoint if the underlying two line segments immediately to the right and to the left of the point meet and form a vertex, whose included angle is statistically larger than a given angle threshold. Statistically, a breakpoint is defined as follows.

Given an observed sequence of ordered points from an arc segment, S as defined above and a point $(\hat{x}_m, \hat{y}_m)$ along the arc segment, the arc point divides the arc segment S into two subsegments S_1 and S_2, where

$$S_1 = \left\{ (\hat{x}_n, \hat{y}_n) \middle| n = 1, \ldots, m \right\}$$

and

$$S_2 = \left\{ (\hat{x}_n, \hat{y}_n) \middle| n = m+1, \ldots, N \right\}.$$

Let $\hat{\theta}_1$ and $\hat{\theta}_2$ be the estimated orientations of the two lines that are fitted to S_1 and S_2, and $\hat{\theta}_{12}$ be the included angle between the lines, $\hat{\theta}_{12}$ is defined as $\hat{\theta}_{12} = \left| \hat{\theta}_1 - \hat{\theta}_2 \right|$, given an included angle threshold θ_0, the breakpoint detection problem may be formulated as a hypothesis testing problem as follows

$$H_0 : \theta_{12} < \theta_0 \qquad H_1 : \theta_{12} \geq \theta_0 \tag{2}$$

• *The authors are with the Intelligent Systems Lab, Department of Electrical Engineering, University of Washington, Seattle, WA 98195-2500. E-mail: {qiangji, haralick}@george.ee.washington.edu.*

Manuscript received 5 May 1997; revised 23 Mar. 1998. Recommended for acceptance by R. Chin.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 106616.

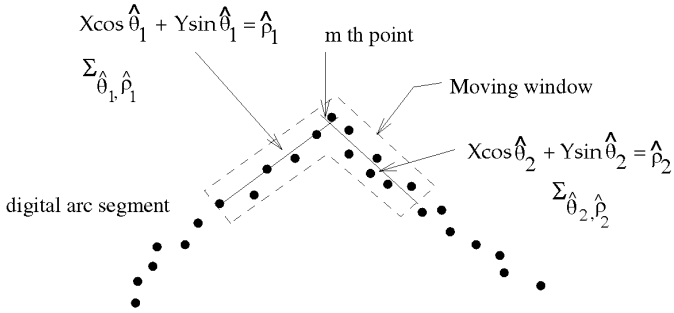

Fig. 1. An example of the breakpoint model.

where θ_{12} represents the population mean of random variable $\hat{\theta}_{12}$.

The hypothesis testing identifies the most likely breakpoint along an arc sequence. If the testing rejects the null hypothesis H_0, it means that the angular orientations of the two lines are statistically different by at least θ_0, and the arc point $(\hat{x}_m, \hat{y}_m)$ is a breakpoint. Fig. 1 illustratively shows the breakpoint model described.

To solve the hypothesis-testing problem, we need to analytically derive the distribution of the random variable $\hat{\theta}_{12}$. To do so, we first need to perform a least-squares line fitting to arc subsegments S_1 and S_2, respectively, resulting in the estimated line orientation parameters $\hat{\theta}_1$ and $\hat{\theta}_2$ for each line. Then we need to analytically estimate the variances σ_1^2 and σ_2^2 for the estimated line parameter $\hat{\theta}_1$ and $\hat{\theta}_2$ using the covariance propagation theory we developed. Finally, we need to derive a test statistic to perform the hypothesis testing. In the section to follow, we will describe the theoretical derivations.

3 Theory for the Proposed Approach

In this section, we detail the theoretical aspects of the developed algorithm. Specifically, we describe least-squares line fitting, covariance propagation, and hypothesis testing.

3.1 Least-Squares Line Fitting

To estimate the line parameters for each arc segment, we perform a least-squares line fitting to the arc points. The least-squares fitting can be formulated as follows.

Assume points $(\hat{x}_n, \hat{y}_n) | n = 1, \ldots, N$, lie on an arc segment S, resulting from perturbation of ideal points (x_n, y_n) locating on the line $x_n\cos\theta + y_n\sin\theta - \rho = 0$. Perturbations are assumed to be independently introduced to each point with Gaussian distributed noise in the direction perpendicular to the line segment as described by the perturbation model in (1).

To estimate the best-fitting line parameters $\hat{\theta}$ and $\hat{\rho}$ using the least-squares method, we minimize the sum of squared residual errors:

$$\epsilon^2 = \sum_{n=1}^{N} \left(\hat{x}_n \cos\hat{\theta} + \hat{y}_n \sin\hat{\theta} - \hat{\rho}\right)^2. \tag{3}$$

In matrix formulation, (3) can be written in quadratic form

$$\epsilon^2 = \hat{\Theta}^t \hat{S} \hat{\Theta}$$

where $\hat{\Theta}$ is the parameter vector and $\hat{S} = D^t D$ is called the scatter matrix. D and $\hat{\Theta}$ are defined as

$$\hat{D} = \begin{pmatrix} \hat{x}_1 & \hat{y}_1 & -1 \\ \hat{x}_2 & \hat{y}_2 & -1 \\ \vdots & \vdots & \vdots \\ \hat{x}_N & \hat{y}_N & -1 \end{pmatrix}$$

and

$$\hat{\Theta} = \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \\ \hat{\rho} \end{pmatrix}$$

where $\hat{\alpha} = \cos\hat{\theta}$ and $\hat{\beta} = \sin\hat{\theta}$.

As a result, we need minimize $\hat{\Theta}^t \hat{S} \hat{\Theta}$ subject to $\hat{\alpha}^2 + \hat{\beta}^2 = 1$. Introducing the Lagrange multiplier λ, the function to be minimized can be expressed as

$$\epsilon^2 = \hat{\Theta}^t \hat{S} \hat{\Theta} - \lambda\left(\hat{\Theta}^t C \hat{\Theta} - 1\right), \tag{4}$$

where C is referred to as constraint matrix and is defined as

$$C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Taking partial derivatives of ϵ^2 with respect to $\hat{\Theta}$ and setting the derivative to zero yields

$$\hat{S}\hat{\Theta} - \lambda C \hat{\Theta} = 0. \tag{5}$$

This system is readily solved by considering generalized eigenvectors of (5). Of the two possible eigenvalues, we select the one that minimizes the sum of geometric distances. Expressed in a quadratic form, the quantity to minimize is

$$\hat{\Theta}^t \hat{S} \hat{\Theta} = \hat{\Theta}^t \lambda C \hat{\Theta} = \lambda.$$

Since $\hat{S}$ is positive definite, the eigenvector $\hat{\Theta} = \left(\hat{\alpha}\,\hat{\beta}\,\hat{\rho}\right)^t$ must correspond to the smaller and positive eigenvalue λ. With $\hat{\alpha}$ and $\hat{\beta}$ determined, the line parameter $\hat{\theta}$ can be estimated from

$$\hat{\theta} = \arctan\left(\hat{\beta}, \hat{\alpha}\right). \tag{6}$$

3.2 Covariance Propagation

The random perturbation on ideal points $X = (x_1, \ldots, x_N, y_1, \ldots, y_N)$, lying on the line $x_n\cos\theta + y_n\sin\theta - \rho = 0$, yields the observed arc points $\hat{X} = \left(\hat{x}_1, \ldots, \hat{x}_N, \hat{y}_1, \ldots, \hat{y}_N\right)$. The use of $\hat{X}$ for estimating line parameter $\Theta = (\theta\,\rho)^t$ yields $\hat{\Theta} = \left(\hat{\theta}\,\hat{\rho}\right)^t$, a least-squares estimate of Θ. Perturbation associated with $\hat{X}$ therefore induces a corresponding perturbation on $\hat{\Theta}$. In this section, we analytically estimate the perturbation of $\hat{\Theta}$, expressed by its covariance matrix $\Sigma_{\Delta\Theta}$, in terms of the covariance matrix Σ_{Δ_X} of $\hat{X}$, through the theory of error propagation.

The nonlinear error estimation and propagation technique studied here is an extension of the standard technique which approximates an nonlinear function by a linear function around a point of interest. The standard technique only works well if the input and output are explicitly related by a function. To solve this problem, Haralick [4] recently proposed a similar linearization technique for propagation of errors from input to output, where input and output are not explicitly related by a given function but are related by minimizing an implicit scaler function $F\left(\hat{X}, \hat{\Theta}\right)$, where $\hat{X}$ represent the input vector and $\hat{\Theta}$ is the output vector. According to the theory, Σ_X, the input covariance matrix, and $\Sigma\Theta$, the output covariance matrix are related by

$$\sum_{\Delta\Theta} = \left(\frac{\partial g(X,\Theta)}{\partial\Theta}\right)^{-1} \left(\frac{\partial g(X,\Theta)}{\partial X}\right)^t \sum_{\Delta X} \left(\frac{\partial g(X,\Theta)}{\partial X}\right) \left[\left(\frac{\partial g(X,\Theta)}{\partial\Theta}\right)^{-1}\right]^t \tag{7}$$

where $g = \frac{\partial F}{\partial\Theta}$.

Following this theory, the scaler function F may be defined as

$$F\left(\hat{\Theta}, \hat{X}\right) = \sum_{n=1}^{N} \left(\hat{x}_n \cos\hat{\theta} + \hat{y}_n \sin\hat{\theta} - \hat{\rho}\right)^2$$

and $g = \left(\frac{\partial F}{\partial \Theta} \; \frac{\partial F}{\partial \rho}\right)^t$.

Letting

$$\mu_x = \frac{1}{N}\sum_{n=1}^{N} x_n;\ \mu_y = \frac{1}{N}\sum_{n=1}^{N} y_n;\ S_x^2 = \sum_{n=1}^{N}\left(x_n - \mu_x\right)^2;$$
$$S_y^2 = \sum_{n=1}^{N}\left(y_n - \mu_y\right)^2;\ S_{xy} = \sum_{n=1}^{N}\left(x_n - \mu_x\right)\left(y_n - \mu_y\right) \tag{8}$$

we can compute

$$\frac{\partial g^{2\times 2}}{\partial \Theta} = \begin{pmatrix} \frac{\partial g}{\partial \theta} \\ \frac{\partial g}{\partial \rho} \end{pmatrix} = \begin{pmatrix} \frac{\partial^2 F}{\partial \theta^2} & \frac{\partial^2 F}{\partial \theta \partial \rho} \\ \frac{\partial^2 F}{\partial \rho \partial \theta} & \frac{\partial^2 F}{\partial \rho^2} \end{pmatrix}$$

where

$$\frac{\partial^2 F}{\partial \theta^2} = 2\left[S_y^2 - S_x^2 + N\left(\mu_y^2 - \mu_x^2\right)\right]$$
$$\cos 2\theta - 4\left(S_{xy} + N\mu_x\mu_y\right)\sin 2\theta + 2N\rho\left(\mu_x\cos\theta + \mu_y\sin\theta\right)$$
$$\frac{\partial^2 F}{\partial \rho^2} = 2N;\ \frac{\partial^2 F}{\partial \theta \partial \rho} = \frac{\partial^2 F}{\partial \rho \partial \theta} = 2N\left(\mu_x\sin\theta - \mu_y\cos\theta\right)$$

and

$$\frac{\partial g^{2N\times 2}}{\partial X} = \underbrace{\begin{pmatrix} \frac{\partial^2 F}{\partial\theta\partial x_1} & \frac{\partial^2 F}{\partial\theta\partial y_1} & \frac{\partial^2 F}{\partial\theta\partial x_2} & \frac{\partial^2 F}{\partial\theta\partial y_2} & \cdots & \frac{\partial^2 F}{\partial\theta\partial x_N} & \frac{\partial^2 F}{\partial\theta\partial y_N} \\ \frac{\partial^2 F}{\partial\rho\partial x_1} & \frac{\partial^2 F}{\partial\rho\partial y_1} & \frac{\partial^2 F}{\partial\rho\partial x_2} & \frac{\partial^2 F}{\partial\rho\partial y_2} & \cdots & \frac{\partial^2 F}{\partial\rho\partial x_N} & \frac{\partial^2 F}{\partial\rho\partial y_N} \end{pmatrix}^t}_{2\times 2N}$$

where

$$\frac{\partial^2 F}{\partial\theta\partial x_n} = 2\left[y_n\cos 2\theta - x_n\sin 2\theta + \rho\sin\theta\right];\ \frac{\partial^2 F}{\partial\rho\partial x_n} = -2\cos\theta$$
$$\frac{\partial^2 F}{\partial\theta\partial y_n} = 2\left[x_n\cos 2\theta + y_n\sin 2\theta - \rho\cos\theta\right];\ \frac{\partial^2 F}{\partial\rho\partial y_n} = -2\sin\theta$$

For the given perturbation model in (1), the input covariance matrix Σ_{Δ_X} is block diagonal and given by

$$\sum_{\Delta X} = \sigma^2 \begin{pmatrix} d & \cdots & 0 & 0 \\ 0 & d & \cdots & 0 \\ \vdots & & & \\ 0 & 0 & \cdots & d \end{pmatrix}_{N\times N} \tag{9}$$

where

$$d = \begin{pmatrix} \cos^2\theta & \sin\theta\cos\theta \\ \sin\theta\cos\theta & \sin^2\theta \end{pmatrix}.$$

Using these expressions, we find that

$$\sigma_\theta^2 = \frac{4\sigma^2\left(S_y^2\cos^2\theta + S_x^2\sin^2\theta - S_{xy}\sin 2\theta\right)}{W^2}$$

$$\sigma_{\theta\rho} = \frac{-4\sigma^2\left[\left(\mu_x\sin\theta - \mu_y\cos\theta\right)\left(S_y^2\cos^2\theta + S_x^2\sin^2\theta - S_{xy}\sin 2\theta\right)\right]}{W}$$

$$\sigma_\rho^2 = \frac{4\sigma^2\left[\left(\mu_x\sin\theta - \mu_y\cos\theta\right)^2\left(S_y^2\cos^2\theta + S_x^2\sin^2\theta - S_{xy}\sin 2\theta\right)\right]}{W} + \frac{4\sigma^2\left[N\left(\left(\mu_x\sin\theta - \mu_y\cos\theta\right)^2 - T\right)^2\right]}{W^2}$$

where

$$T = \frac{S_y^2 - S_x^2}{N}\cos 2\theta - \frac{2S_{xy}}{N}\sin 2\theta - \left(\mu_y\sin\theta + \mu_x\cos\theta\right)^2 + \rho\left(\mu_y\sin\theta + \mu_x\cos\theta\right)$$

$$W = 2NT.$$

A further simplification can be performed on the above covariance matrix as follows using the following definition. Define

$$k = \begin{cases} +\sqrt{x^2 + y^2 - \rho^2} & \text{if } y\cos\theta \geq x\sin\theta \\ -\sqrt{x^2 + y^2 - \rho^2} & \text{otherwise} \end{cases}$$

and let

$$\mu_k = \frac{1}{N}\sum_{n=1}^{N} k_n;\ S_k^2 = \sum_{n=1}^{N}\left(k_n - \mu_k\right)^2.$$

After algebraic operations and simplifications, we obtain

$$\sum_{\Delta\Theta} = \begin{pmatrix} \sigma_\theta^2 & \sigma_{\theta\rho} \\ \sigma_{\theta\rho} & \sigma_\rho^2 \end{pmatrix} = \sigma^2 \begin{pmatrix} \frac{1}{S_k^2} & \frac{\mu_k}{S_k^2} \\ \frac{\mu_k}{S_k^2} & \frac{1}{N} + \frac{\mu_k^2}{S_k^2} \end{pmatrix}. \tag{10}$$

Geometrically, k can be interpreted as the signed distance between a point (x, y) and the point on the line closest to the origin. Hence, S_k^2 represents the spread of points along the line. From (10), it is clear that with a larger S_k^2, i.e., points with larger spread along the line, we can obtain better fit as indicated with smaller trace of the covariance matrix. In addition, μ_k is the mean position of the points along the line. It acts like a moment arm. A larger μ_k, i.e., a longer moment arm, can induce more variance to the estimated $\hat{\rho}$.

Equation (10) is significant. It reveals that the covariance matrix of the estimated line parameters not only relates to the input perturbation and to the number of points used, but also relates to the configuration of these points. It shows that points further from the origin may induce more uncertainty to the estimated parameter $\hat{\rho}$. It further indicates that points with larger spread along the fitted line yield smaller perturbation on the estimated parameters. These relations are significant in that we can always translate the coordinate system to minimize the perturbations on the estimated line parameters and therefore to improve the quality of the estimated line parameters. Equation (10) also reveals that the estimated parameters $\hat{\theta}$ and $\hat{\rho}$ are correlated unless $u_k = 0$ and that $\hat{\rho}$ tends to incur more perturbation than $\hat{\theta}$. Further investigation of (10) reveals that σ_θ^2 is invariant to coordinate translation and rotation while σ_ρ^2 is variant to coordinate translations that change μ_k, contrary to the popular belief that σ_ρ^2 is invariant.

The way in which we have derived the covariance matrix $\Sigma_{\Delta\Theta}$ requires unperturbed and unknown input X and ideal parameter Θ. We can obtain an estimate of $\Sigma_{\Delta\Theta}$ by substituting X by the observed input $\hat{X}$ and Θ by the estimated parameter $\hat{\Theta}$. In addition, σ^2 is estimated from the sum of the residual errors, assuming the

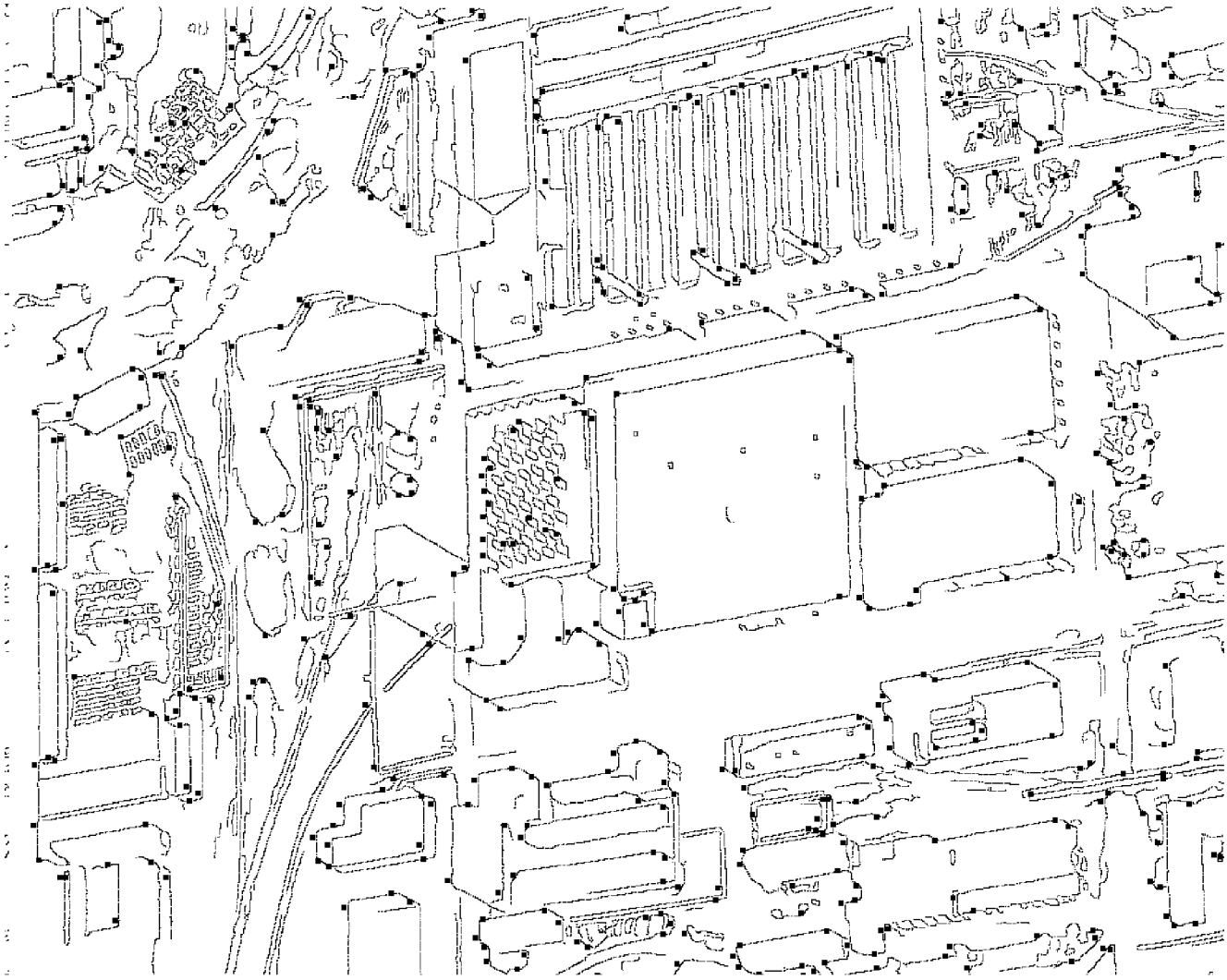

Fig. 2. Extracted edges of building model for a RADIUS image with detected corners (represented by black square dots) overlaid.

residual error for each point has a Gaussian distribution with zero mean and a variance of σ^2.

3.3 Hypothesis Testing

With the covariance matrix computed, we can proceed to develop a test statistic to decide whether the estimated angular parameters of the two fitted lines differ by a threshold θ_0. Given two arc segments S_1 and S_2, a least-squares line fitting is performed to fit a line to S_1 and to fit a line to S_2, respectively, using the method described in Section 3.1, thus resulting in the estimated line orientation parameters $\hat{\theta}_1$ and $\hat{\theta}_2$. From (10), we obtain $\sigma^2_{\theta_1}$ and $\sigma^2_{\theta_2}$, the estimated variances of $\hat{\theta}_1$ and $\hat{\theta}_2$. The hypothesis testing can then be formulated as

$$H_0: \theta_{12} < \theta_0 \qquad H_1: \theta_{12} \geq \theta_0 \tag{11}$$

where θ_0 (ranging from zero to 90 degrees) is a user-supplied angular threshold and θ_{12} is the population mean of random variable $\hat{\theta}_{12}$, which is defined as

$$\hat{\theta}_{12} = \left|\hat{\theta}_1 - \hat{\theta}_2\right|$$

since

$$\hat{\theta}_1 \sim N\left(\theta_1, \sigma^2_{\theta_1}\right) \text{ and } \hat{\theta}_2 \sim N\left(\theta_2, \sigma^2_{\theta_2}\right).$$

Thus, a likelihood ratio test statistic can be designed as follows

$$T = \frac{\hat{\theta}^2_{12}}{\sigma^2_{\theta_1} + \sigma^2_{\theta_2}}. \tag{12}$$

The distribution of the test statistic T under null hypothesis is a noncentral chi-squared with two degrees of freedom, and the noncentrality parameter d are

$$T \sim \chi^2_2 \text{ and } d = \frac{\theta^2_0}{\sigma^2_{\theta_1} + \sigma^2_{\theta_2}}.$$

Given the test statistic and its distribution, an appropriate significant level α may be selected to perform the test. If the p-value[1] of a test is larger than α, the null hypothesis is accepted, i.e., no breakpoint exists between S_1 and S_2. On the other hand, if the p-value of the test is less than α, the null hypothesis is rejected, and the vertex formed by arcs S_1 and S_2 is declared a breakpoint.

3.4 Breakpoint Optimization

The set of breakpoints detected from a digital arc are only optimal locally but not globally. This may result in locational errors with

1. The p-value of a test statistic x at x_0 is defined as $prob(x > x_0 \mid H_0)$, where H_0 is the null hypothesis.

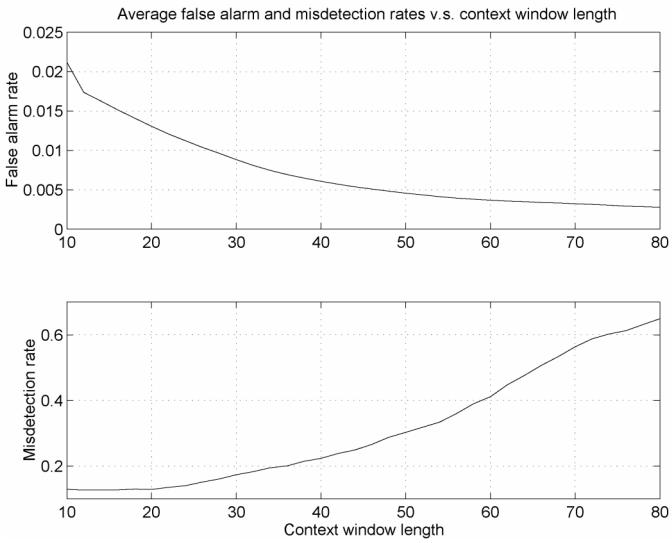

Fig. 3. Misdetection and false alarm rates versus context window length, $\theta_0 = 30$.

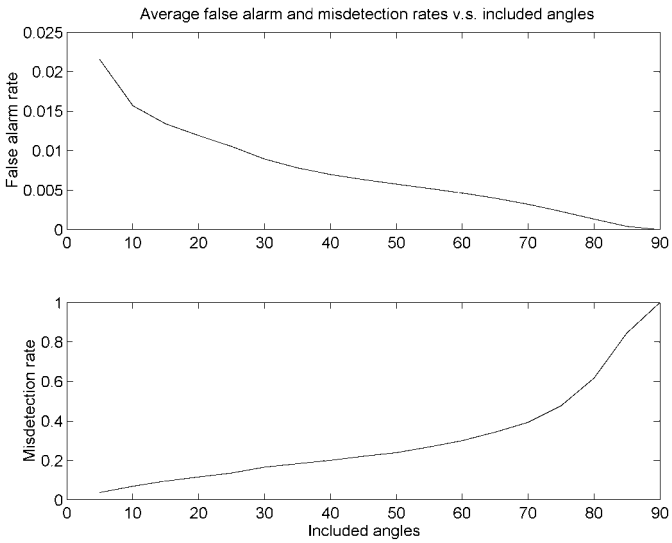

Fig. 4. Misdetection and false alarm rates versus included angle threshold, with window length being 25 pixels.

detected breakpoints. To reduce the location errors with the detected breakpoints, we perform a breakpoint optimization for all breakpoints detected on a digital arc sequence. The breakpoint optimization, based on Pavlidis's discrete optimization method [9], iteratively alternate-shifts all the detected breakpoints until breakpoint shifting does not improve the error. Pavlidis suggested several error norms to use. For this research, the error norm we used is the variance of the residual fitting errors as defined by (3).

4 Performance Characterization

This section covers the performance analysis and characterization of the breakpoint detector using images from our RADIUS images [15]. A total of 80 model board images were used. They are divided into three sets: J, K, and M, taken from two model-boards, with different orientations. Each image represents an outdoor scene, containing primarily building structures. The breakpoint detector

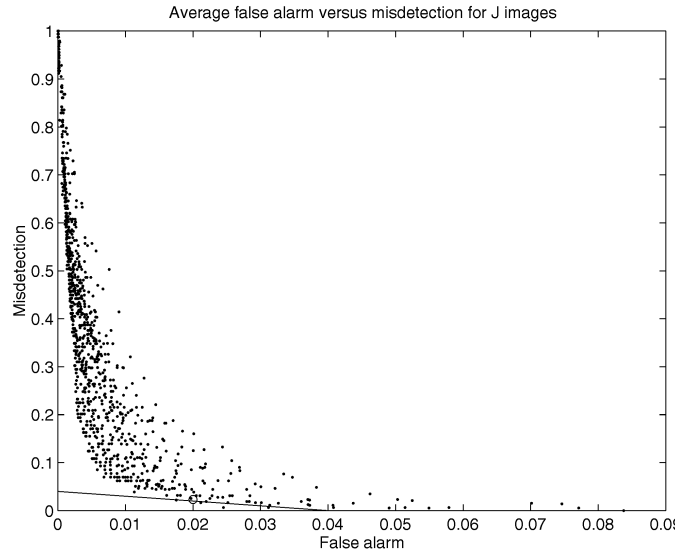

Fig. 5. ROC analysis of the breakpoint detector for *J* images.

was applied to detect building corners.[2] The criteria used for the evaluation are misdetection and false alarm rates. The groundtruth data used for computing the misdetection and false alarm rates are obtained by manually annotating the aerial images to delineate the edges of the buildings and other structures in the image. All vertices formed by the line segments are selected as the groundtruth corners [15]. In addition, we perform a receiver operating characteristics (ROC) analysis of the breakpoint detector to identify the optimal operating parameters for the RADIUS images (see Fig. 2).

Fig. 3 plots the average false alarm (top) and misdetection (bottom) rates versus the context window length for all 27 *J* images. Increasing window length leads to a decrease in false alarm rate but an increase in misdetection. The decrease in false alarm rate may be partially due to the fact that larger context window yields more data for more accurate statistical analysis. The decrease tapers off, however, for large context window. This shows that once a threshold is reached, increasing the window length only yields marginal improvement to false alarm rate. On the other hand, an increase in misdetection rate as the window size increases, may be due to the assumption that only one breakpoint is allowed to be present within the context window. Increasing window size leads to the presence of multiple breakpoints in the window, this will lead to an increase in σ_θ^2, causing the test statistic to be insensitive to orientation difference, thus increasing the misdetection rate.

Fig. 4 plots the average false alarm rate (top) and misdetection rate (bottom) versus the included angle threshold for all *J* images. It shows as the included angle threshold increases, the false alarm rate tends to decrease while the misdetection rate increases. Increasing the included angles removes breakpoints with small included angles, therefore reducing the false alarm rate. On the other hand, the removed breakpoints may contain true breakpoints, therefore increasing the misdetection rate. Fig. 4 also shows that the included angle threshold has a much stronger impact on misdetection than on false alarm. While a small increase in the angle threshold leads to marginal improvement in false alarm, it could lead to a dramatic increase in misdetection.

The result of ROC analysis is shown in Fig. 5, which plots average probability of false alarm as a function of the average probability of misdetection for all *J* images. Each point in Fig. 5 results from a particular combination of a context window length and an

2. Building corners refer to corners of man-made structures, with orthogonal adjacent surfaces.

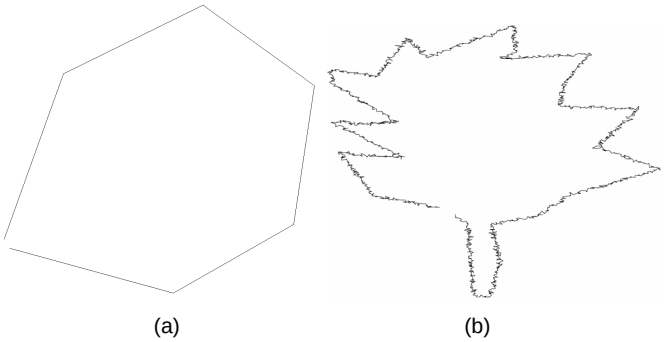

Fig. 6. Two synthetic curves used for comparison.

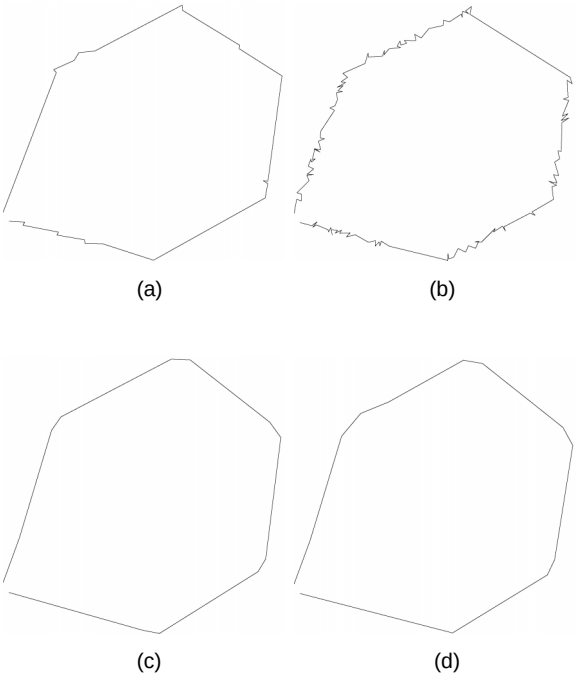

Fig. 7. Polygonal approximation of the perturbed test curves shown in Fig. 6a. (a) and (b) From Lowe's algorithm. (c) and (d) From our algorithm with context window length = 30 and included angle threshold = 5. (a) and (c) are perturbed with $\sigma = 0.5$, and (b) and (d) are perturbed with $\sigma = 1.5$.

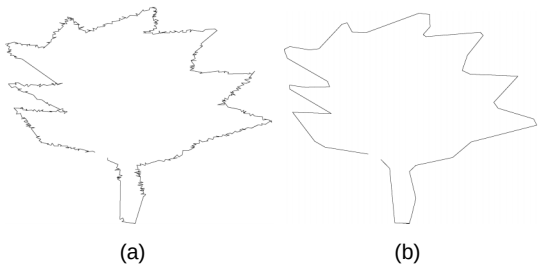

Fig. 8. Results of polygonal approximation to the curves shown in Fig. 6b using Lowe's algorithm (a) and our algorithm (b), where window length = 50 and angular threshold = 25.

included angle threshold. The lower boundary of the ROC plot (referred to as ROC curve) represents the optimal operating range for the breakpoint detector. The optimal operating parameters can be determined from the tangent intersection point between the ROC curve and a line representing the cost function. The line shown in Fig. 5 represents a cost function of equal effective cost[3] for committing a misdetection and false alarm. It is clear that at the optimal point (denoted by the small circle), the misdetection rate is about 2.5 percent while the false alarm rate is about 2.2 percent. The optimal operating point can be achieved with a window length of and angular threshold of 30 and five, respectively.

Given the optimal context window length (30) and included angle threshold (five) derived from the ROC analysis, we applied the breakpoint detector to all 80 RADIUS images. The results show that the breakpoint detector has an average misdetection rate of less than 2.3 percent and false alarm rate of about 2.1 percent, respectively. The 2.1 percent false alarm rate corresponds to about 20 false alarm points per 1,000 arc points. This result echos the one obtained from the ROC analysis. Fig. 2 gives an example of the detected corners overlaid on the corresponding edge image of a RADIUS image, where the detected corners are represented by small black dots.

5 Performance Comparison

This section describes results of evaluating the performance of our breakpoint detector against that of Lowe's algorithm [7] using both synthetic data and RADIUS data.[4] Lowe's algorithm for digital arc segmentation has been widely cited and was found superior to most breakpoint detection algorithms available [13], [12], [5].

5.1 Synthetic Curves

First, we evaluate the performance of Lowe's algorithm and ours using synthetic curves for polygonal approximation. Synthetic curves were generated by sampling the original model curve consisting of piecewise linear line segments and by perturbing each sampled pixel with iid Gaussian noise with mean zero and variance σ^2. The reconstructed test curves consist of perturbed sampled points. Fig. 6a shows a model curve before corruption. Fig. 6b shows a perturbed synthetic curve adapted from Teh [14]. The model curve in Fig. 6a was corrupted with two different noise levels ($\sigma = 0.5$) and ($\sigma = 1.5$), respectively, to which the two breakpoint detectors are applied. Results from Lowe's algorithm are shown in Fig. 7a and Fig. 7b, while results from our algorithm are shown in Fig. 7c and Fig. 7d. It is clear from Fig. 7 that while both algorithms can detect all breakpoints (five breakpoints from the model curve), our algorithm is superior to Lowe's algorithm in that Lowe's algorithm tends to detect local irregular arc points as breakpoints, therefore yielding a much higher false alarm rate than our algorithm. Furthermore, increasing the noise level leads to an increase in the false alarm rate on both algorithms but its impact on our algorithm is much smaller than on Lowe's algorithm.

To further compare the two algorithms, we applied them to the curve shown in Fig. 6b. The results from Lowe's algorithm and our algorithm are shown in Fig. 8a and Fig. 8b, respectively. Visually, both algorithms performed equally well on the two curves, but our algorithm outperformed Lowe's algorithm. Lowe's algorithm tends to detect local irregularity like small bumps or dips as breakpoints, therefore yielding a high false alarm rate.

5.2 RADIUS Data

A detailed comparison was also carried out on four different RADIUS images. The goal here is to find the building corners from the edge images of the buildings. Groundtruth building corners were obtained via the annotation procedure as described above. The

3. Effective cost is the actual cost of a mistake multiplied by its probability.

4. The source code for Lowe's algorithm can be obtained from ftp://ftp.brunel.ac.uk/CompSci/Paul.Rosin/curves.

TABLE 1
PERFORMANCE OF LOWE'S ALGORITHM

Images	FA	MD
1	10.6	1.3
2	9.2	1.6
3	9.3	0.7
4	10.1	3.8

TABLE 2
PERFORMANCE OF OUR ALGORITHM

Images	FA	MD
1	2.5	1.9
2	5.4	1.8
3	1.3	1.8
4	4.4	2.0

criteria are the misdetection and false alarm rates. Table 1 and Table 2 quantitatively show the performance of the two algorithms for each of the four images. As expected, while both algorithms are comparable in terms of misdetection rates, our algorithm has a much lower false alarm rate for all four images. Lowe's algorithm tends to make two to seven times as many false alarm mistakes. This again demonstrates the superiority of our algorithm.

6 DISCUSSION AND CONCLUSIONS

In this paper, we have presented a statistical approach for detecting breakpoints on digital arc sequences. A breakpoint is defined to be an arc point where two line segments meet and form a vertex. The arc point closest to the vertex point is declared as a breakpoint if the angular orientations of the two lines that form the vertex are *significantly statistically different*. The essence of the breakpoint detector lies in the covariance propagation theory we developed for propagating input perturbation to the perturbation of estimated line parameters. Based on the analytically derived covariance matrix, a test statistic is designed to statistically check each arc point.

The performance of the breakpoint detection technique was evaluated using 80 images from RADIUS dataset. A comparative performance evaluation was also conducted using both synthetic data and RADIUS data. The study showed that the algorithm is robust and accurate for the complex images. It has an average misdetection rate of 2.3 percent and false alarm rate of 2.1 percent for the complex RADIUS images. The comparative study revealed that our algorithm consistently outperforms Lowe's technique on both synthetic and real data. While our algorithm has a comparable misdetection rate with existing methods, our method yields a much lower false alarm rate. This is because the technique has great statistical efficiency and it properly takes into account the perturbation on the estimated line parameters.

Our technique, however, has the following limitations. First while the parameters α and θ_0 are scale invariant, the context window length is not scale invariant. Given different scales of the same image, the context window length may need to be tuned separately to achieve optimal performance. Second, for reliable statistical testing, the window size cannot be too small. Our study shows that window size less than six pixels may not yield reliable breakpoints. Third, the technique is not effective in detecting breakpoints formed by very curved line segments like a circle. Finally, like all breakpoint detection techniques, our technique is limited by the performance of the edge tracing and linking methods employed to produce the digital curves which are input to breakpoint detection technique.

ACKNOWLEDGMENTS

This work was partially supported by a grant from the Boeing Commercial Airplanes.

REFERENCES

[1] J.G. Dunham, "Optimal Uniform Piecewise Linear Approximation of Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 67-75, 1986.
[2] H. Freeman and L.S. Davis, "A Corner-Finding Algorithm for Chain-Coded Curves," *IEEE Trans. Computers*, vol. 26, pp. 297-303, 1977.
[3] R. Guiducci and S. Nichani, "Corner Detection," *Pattern Recognition*, vol. 23, no. 11, pp. 1,223-1,233, 1990.
[4] R.M. Haralick, "Propagating Covariance in Computer Vision," *Proc. 12th ICPR*, pp. 493-498, 1994.
[5] T. Kadonaga and K. Abe, "Comparison of Methods for Detecting Corner Points From Digital Curves," *Int'l Workshop Graphics Recognition*, 1995.
[6] L. Kitchen and A. Rosenfeld, "Gray Level Corner Detection," *Pattern Recognition Letters*, pp. 95-102, 1982.
[7] D.G. Lowe, "Three Dimensional Object Recognition From Single Two Dimensional Images," *Artificial Intelligence*, vol. 31, pp. 355-395, 1987.
[8] L. O'Gorman, "Curvilinear Feature Detection From Curvature Estimation," *Proc. Ninth Int'l Conf. Pattern Recognition*, 1988.
[9] T. Pavlidis, "Waveform Segmentation Through Functional Approximation," *IEEE Trans. Computers*, vol. 22, no. 7, p. 689, 1973.
[10] T. Pavlidis and S.L. Horowitz, "Segmentation of Plane Curves," *IEEE Trans. Computers*, vol. 23, no. 8, p. 860, 1974.
[11] A. Rosenfeld and J.S. Weszka, "An Improved Method of Angle Detection on Digital Curves," *IEEE Trans. Computers*, vol. 24, pp. 940-941, 1975.
[12] P.L. Rosin, "Techniques for Assessing Polygonal Approximations of Curves," *Seventh British Machine Vision Conf.*, Edinburgh, 1996.
[13] P.L. Rosin and G.A.W. West, "Non-Parametric Segmentation of Curves Into Various Representations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 12, pp. 1,140-1,153, Dec. 1995.
[14] C.H. Teh and R.T. Chin, "On the Detection of Dominant Points in Digital Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, pp. 859-872, 1989.
[15] K. Thornton and et al., "Groundtruthing the RADIUS Model-Board Imagery," *IUE Workshop*, 1995.
[16] J.A. Ventura and J.M. Chen, "Segmentation of Two-Dimensional Curve Contours," *Pattern Recognition*, vol. 25, no. 10, pp. 1,129-1,140, 1992.
[17] X. Zhang, R.M. Haralick, R. Ramesh, and A.S. Bedekar, "A Bayesian Corner Detector: Theory and Performance Evaluation," *IUE Workshop*, 1995.
[18] O.A. Zuniga, "Corner Detection Using the Facet Model," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 30-37, 1983.