Corner Detection with Covariance Propagation

Qiang Ji and Robert M. Haralick
Intelligent Systems Laboratory
Department of Electrical Engineering, FT-10
University of Washington
Seattle, WA 98195

Abstract

This paper presents a statistical approach for detecting corners from chain encoded digital arcs. An arc point is declared as a corner if the estimated parameters of the two fitted lines of the two arc segments immediately to the right and left of the arc point are statistically significantly different. The corner detection algorithm consists of two steps: corner detection and optimization. While corner detection involves statistically identifying the most likely corner points along an arc sequence, corner optimization deals with improving the locational errors of the detected corners.

The major contributions of this research include developing a method for analytically estimating the covariance matrix of the fitted line parameters and developing a hypothesis test statistic to statistically test the difference between the parameters of two fitted lines. Performance evaluation study showed that the algorithm is robust and accurate for complex images. It has an average misdetection rate of 2.5% and false alarm rate of 2.2% for the complex RADIUS images. This paper discusses the theory and performance characterization of the proposed corner detector.

1. Introduction

Corners have long been important two dimensional features for computer vision research. They have been used extensively for matching, pattern recognition, and data compression. Various algorithms have been developed for detecting corners. Corner detection algorithms can be roughly grouped into two categories: one is based on the detection directly from the underlying grayscale images; the other is based on the digital arcs, resulting from edge detection and linking. Various techniques have been developed for corner 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 corner points including maximum curvature, deflection angle, maximum deviation, and total fitting errors [1][5][6][11]. A major problem with existing approaches is that the employed criterion is not tied to a statistical analysis, therefore rendering existing methods susceptible to noise. To overcome this, we present a statistical approach for corner detection. Here, the corner criterion is treated as a random variable and is subject to perturbation. Given an arc segment, a corner is defined to be a point where two underlying line segments meet and form a vertex, whose included angle is statistically larger than an angle threshold. The corner detection procedure involves sliding a context window of specified length over the arc sequence, performing a least square line fitting to the arc points located immediately to the left and right of the center of the window, estimating the parameters of the fitted lines, and their covariance matrices, and finally performing a hypothesis test to test the statistical difference between the parameters of the two fitted lines. If the difference is significant, the arc point located in the center of the window is declared as a corner point. Finally, a corner optimization procedure is performed to improve the locational errors of the detected corners.

The major contributions of this research include developing a method for analytically estimating the covariance matrix of the fitted line parameters and developing a hypothesis test statistic to statistically test the difference between the parameters of two fitted lines. This paper is arranged as follows. In section 2, we state the problem and present the associated noise and corner models. Section 3 discusses in detail the theoretical aspects of the corner detector. The performance evaluation of the corner detector is covered in sections 4 and 5. The paper ends in section 6 with a discussion and summary of the proposed approach.

2. Problem statement

A corner point represents a discontinuity in the curvature of a curve. The location of the discontinuity can be

1063-6919/97 $10.00 © 1997 IEEE

Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)
1063-6919/97 $10.00 © 1997 IEEE

approximated by the intersection of two straight lines that underlie the arc segments to the right and left of the corner point. Perturbation to the points on the ideal underlying lines gives rise to the observed arc segments. This section is concerned with the definition of the perturbation model and the corner model.

2.1. Perturbation model

Given an observed sequence of ordered points from a line arc segment, $S = \{(\hat{x}_n, \hat{y}_n) | n = 1, \ldots, N\}$, where N is the number of points on the arc segment, the perturbation model assumes that $(\hat{x}_n, \hat{y}_n)$ result from random perturbations to the ideal points (x_n, y_n), $n = 1, \ldots, N$, 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:

$$\begin{pmatrix} \hat{x_n} \\ \hat{y_n} \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \end{pmatrix} + \xi_n \begin{pmatrix} \cos\theta \\ sin\theta \end{pmatrix} \quad (1)$$

where $n = 1, \ldots, N$ and ξ_n are independently and identically distributed as N(0, σ^2).

2.2. Corner model

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

Given an observed sequence of ordered points from an arc segment, $S = \{\begin{pmatrix} \hat{x}_n \\ \hat{y}_n \end{pmatrix} | n = 1, \ldots, N\}$ and a point $(\hat{x}_k, \hat{y}_k)$ along the arc segment, the arc point divides the arc segment S into two sub-segments S_1 and S_2 , where $S_1 = \{\begin{pmatrix} \hat{x}_n \\ \hat{y}_n \end{pmatrix} | n = 1, \ldots, k\}$ and $S_2 = \{\begin{pmatrix} \hat{x}_n \\ \hat{y}_n \end{pmatrix} | n = k+1, \ldots, N\}$. Let $\hat{\theta}_1$ and $\hat{\theta}_2$ be the estimated orientations of the two lines that fit to S_1 and S_2, and $\hat{\theta}_{12}$ be the included angle between the lines, $\hat{\theta}_{12}$ is then defined as

$$\hat{\theta}_{12} = |\hat{\theta}_1 - \hat{\theta}_2| \quad (2)$$

Given an included angle threshold θ_0, the corner detection problem may be formulated as a hypothesis testing problem as follows:

$$H_0 : \theta_{12} < \theta_0 \qquad H_1 : \theta_{12} \geq \theta_0 \quad (3)$$

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

The hypothesis testing identifies the most likely corner point along an arc sequence. Specifically, given a significant level α, the P-value of each observed $\hat{\theta}_{12}$ is computed and compared with α. If the P-value $\leq \alpha$, the null hypothesis is rejected and the arc point being considered $(\hat{x}_k, \hat{y}_k)$ is a corner point. Figure 1 illustratively shows the corner model just described. In the section to follow, we describe the theoretical derivations that lead to the solution to the above hypothesis testing problem.

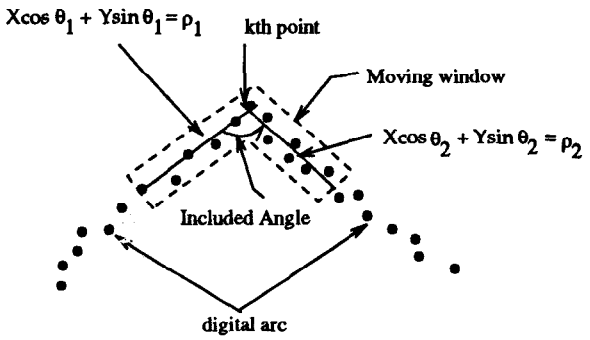

Figure 1. An example of the corner model

3. Theory for the proposed approach

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

3.1. Least-square line fitting

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

Assume points $(\hat{x}_n, \hat{y}_n)$, $n = 1...N$, lie on an arc segment S, resulting from perturbation of ideal points (x_n, y_n) locating on the line $x_n \cos\theta + y_n \sin\theta - \rho = 0$. Perturbation to each point follow the perturbation model in equation (1).

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

$$\begin{aligned} \epsilon^2 &= ||\hat{D}\hat{\Theta}||^2 \\ &= \hat{\Theta}^t \hat{D}^t \hat{D} \hat{\Theta} \\ &= \hat{\Theta}^t \hat{S} \hat{\Theta} \end{aligned} \quad (4)$$

Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)
1063-6919/97 $10.00 © 1997 IEEE

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

$$\hat{D} = \begin{pmatrix} \hat{x}_1 & \hat{y}_1 & -1 \\ \hat{x}_2 & \hat{y}_2 & -1 \\ \vdots & \vdots & \vdots \\ \hat{x}_N & \hat{y}_N & -1 \end{pmatrix} \text{ and } \hat{\Theta} = \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \\ \hat{\rho} \end{pmatrix}$$

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

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

$$\epsilon^2 = \hat{\Theta}^t\hat{S}\hat{\Theta} - \lambda(\hat{\Theta}^tC\hat{\Theta} - 1) \tag{5}$$

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 w.r.t. $\hat{\Theta}$ and λ, and setting them to zeros yields the simultaneous equations

$$\hat{S}\hat{\Theta} - \lambda C\hat{\Theta} = 0 \tag{6}$$

$$\hat{\Theta}^tC\hat{\Theta} = 1 \tag{7}$$

This system is readily solved by considering generalized eigenvectors of eq(6).

3.2. Covariance propagation

The random perturbation on ideal points X=(x_n, y_n), $n = 1,\ldots,N$, lying on line $x_n\cos\theta + y_n\sin\theta - \rho = 0$, yields observed arc points $\hat{X} = (\hat{x}_n, \hat{y}_n)$, $n = 1,\ldots,N$. The use of $\hat{X}$ for estimating line parameter $\Theta = \begin{pmatrix} \theta \\ \rho \end{pmatrix}$ yields $\hat{\Theta} = \begin{pmatrix} \hat{\theta} \\ \hat{\rho} \end{pmatrix}$, a least square estimate of Θ. The perturbation accompanying $\hat{X}$ induces a corresponding perturbation on $\hat{\Theta}$. In this section, we will analytically estimate $\Delta\Theta$, the perturbation of $\hat{\Theta}$, expressed in its covariance matrix $\Sigma_{\Delta\Theta}$, in terms of the covariance matrix $\Sigma_{\Delta X}$ of $\hat{X}$.

Based on the covariance propagation theory [2], the scaler criterion function F that needs to be minimized can be defined as

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

Define g as follows:

$$g^{2\times1} = \frac{\partial F}{\partial\Theta} = \begin{pmatrix} \frac{\partial F}{\partial\theta} \\ \\ \frac{\partial F}{\partial\rho} \end{pmatrix} \tag{8}$$

Then based on [2], $\Sigma_{\Delta\Theta}$, the covariance matrix of the estimated line parameters $\hat{\Theta}$ can be computed from:

$$\begin{aligned} \sum_{\Delta\Theta} &= \begin{pmatrix} \sigma_\theta^2 & \sigma_{\theta\rho} \\ \sigma_{\theta\rho} & \sigma_\rho^2 \end{pmatrix} \\ &= (\frac{\partial g(X,\Theta)}{\partial\Theta})^{-1}(\frac{\partial g(X,\Theta)}{\partial X})^t\sum_{\Delta X} \\ &\quad (\frac{\partial g(X,\Theta)}{\partial X})[(\frac{\partial g(X,\Theta)}{\partial\Theta})^{-1}]^t \end{aligned} \tag{9}$$

where $\frac{\partial g}{\partial X}$ and $\frac{\partial g}{\partial\Theta}$ are evaluated at ideal line parameter Θ and points X, and $\Sigma_{\Delta X}$ represents the input perturbation. From eq(9), we can easily obtain

$$\frac{\partial g^{2\times2}}{\partial\Theta} = \begin{pmatrix} \frac{\partial g}{\partial\theta} \\ \\ \frac{\partial g}{\partial\rho} \end{pmatrix} = \begin{pmatrix} \frac{\partial^2 F}{\partial\theta^2} & \frac{\partial^2 F}{\partial\theta\partial\rho} \\ \\ \frac{\partial^2 F}{\partial\rho\partial\theta} & \frac{\partial^2 F}{\partial\rho^2} \end{pmatrix}$$

and,

$$\frac{\partial g^{2N\times2}}{\partial X} = \underbrace{\begin{pmatrix} \frac{\partial^2 F}{\partial\theta\partial x_1} & \frac{\partial^2 F}{\partial\rho\partial x_1} \\ \frac{\partial^2 F}{\partial\theta\partial y_1} & \frac{\partial^2 F}{\partial\rho\partial y_1} \\ \vdots & \vdots \\ \frac{\partial^2 F}{\partial\theta\partial x_N} & \frac{\partial^2 F}{\partial\rho\partial x_N} \\ \frac{\partial^2 F}{\partial\theta\partial y_N} & \frac{\partial^2 F}{\partial\rho\partial y_N} \end{pmatrix}}_{2N\times2}$$

For the given perturbation model in equation (1), the input covariance matrix $\Sigma_{\Delta X}$ is given by

$$\Sigma_{\Delta X} = \sigma^2\begin{pmatrix} d & \ldots & 0 & 0 \\ 0 & d & \ldots & 0 \\ \vdots & & & \\ 0 & 0 & \ldots & d \end{pmatrix}_{N\times N}$$

where

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

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

$$\mu_k = \frac{1}{N}\sum_{n=1}^{N}k_n$$

$$S_k^2 = \sum_{n=1}^{N}(k_n - \mu_k)^2$$

Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)
1063-6919/97 $10.00 © 1997 IEEE

After algebraic operations and simplifications, we obtain

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

Geometrically, k can be interpreted as the signed distance between a point (x,y) and the point on the line closest to the origin. As a result, S_k^2 represents the spread of points along the line. A larger S_k^2, i.e., points with larger spread along the line yields better fit as indicated with smaller covariance matrix trace. 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 $\hat{\rho}$. Further investigation of eq(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.

The way in which we have derived the covariance matrix $\Sigma_{\Delta\Theta}$ requires that the matrices $\frac{\partial g(X,\Theta)}{\partial\Theta}$ and $\frac{\partial g(X,\Theta)}{\partial X}$ be known. But X and Θ are not observed. $\hat{X}$ and $\hat{\Theta}$ are observed instead. So, to obtain an estimated covariance matrix $\hat{\Sigma}_{\Delta\Theta}$, we substitute $\hat{X}$ and $\hat{\Theta}$ for X and Θ in eq(10).

3.3. Hypothesis testing

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

$$H_0 : \theta_{12} < \theta_0 \qquad H_1 : \theta_{12} \geq \theta_0 \quad (11)$$

where θ_0 (ranging from 0 to 90 degree) is a user supplied angular threshold, and θ_{12} is the population mean of random variable $\hat{\theta}_{12}$, which is defined as

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

since

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

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

$$T = \frac{\hat{\theta}_{12}^2}{\hat{\sigma}_{\theta_1}^2 + \hat{\sigma}_{\theta_2}^2} \quad (12)$$

The distribution of the test statistic under null hypothesis is a non-central Chi-squared with two degrees of freedom.

$$T \sim \chi_2^2$$

where the noncentrality parameter h is

$$h = \frac{\theta_0^2}{\hat{\sigma}_{\theta_1}^2 + \hat{\sigma}_{\theta_2}^2}$$

Given the test statistic and its distribution, a significant level of $\alpha = 0.01$ was selected to perform the test. If the p-value of a test is larger than α, the null hypothesis is accepted, i.e., no corner 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 corner.

3.4. Corner optimization

The set of corner points detected in a digital arc are only optimal locally but not globally. It is not globally optimal because not all the points on the arc are used to detect the corner points. This may result in high locational errors. To reduce the location errors with the detected corners, we perform a corner optimization. The corner optimization, based on Pavlidis's discrete optimization method [5], iteratively shifts the detected corner points to produce a better approximation of the arc sequence. While the iterative optimization procedure is guaranteed to terminate with improved location errors, it however may terminate at a local minimum rather than at the global minimum.

4. Performance evaluation

This section discusses results from a series of experiments aimed at characterizing the performance of the proposed algorithm using images from the RADIUS database. A total of 80 model board images are used. Each image represents an outdoor scene, containing primarily building structures. The input to the corner detector are sequences of arc segments resulting from an edge detection and linking operation. The groundtruth points for these images are obtained by manually annotating the aerial images to delineate the edges of the buildings and other structures in the image [10]. The criteria used for the evaluation are misdetection (MD) and false alarm (FA) rates.

Figure 2 plots the average false alarm rate and misdetection rate versus the context window length for all images. It shows as the window length increases, the false alarm rate decreases quadratically while the misdetection rate increases quadratically. A larger context window yields more data for more accurate statistical analysis, which leads to

Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)
1063-6919/97 $10.00 © 1997 IEEE

a decrease in false alarm rate. On the other hand, the assumption of only one corner present in the context window leads to an increase in misdetection as window size increases. However, this increase only becomes apparent after the window size exceeds certain threshold.

Figure 3 gives the average MD and FA rates versus the included angle threshold. It shows as the included angle increases, the false alarm rate tends to decrease. This is because we are looking for building corners, which often have large included angles. Increasing the included angles filters out corners with small included angles, therefore reducing the false alarm rate. On the other hand, increasing the included angle may increase the misdetection rate. Figure 3 also shows that while a small increase in the angle threshold leads to marginal improvement in false alarm, it however could lead to a dramatic increase in misdetection.

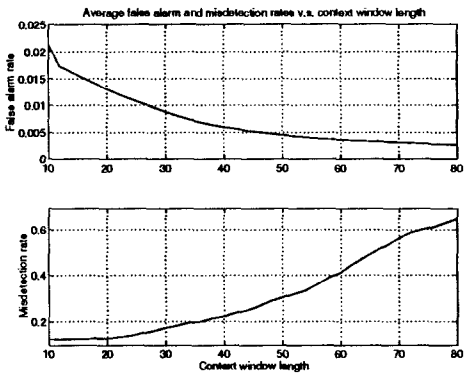

Figure 2. Misdetection and false alarm rates versus context window length, $\theta_0 = 30$

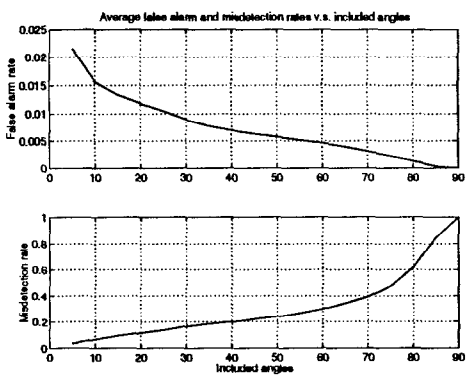

Figure 3. Misdetection and false alarm rates versus included angle threshold, with window length being 25 pixels

We also study the average performance of the corner detector for all 80 RADIUS images using the context window length and included angle thresholds derived from previous experiments, i.e., optimal window length of 30 pixels and optimal included angle threshold of 5 degree. The results show that the corner detector has an average misdetection rate of about 2.5% and false alarm rate of about 2.2% respectively.

5. Performance comparison

This section describes results of evaluating the performance of our corner detector against that of Lowe's algorithm [4] using both synthetic data and RADIUS data. The criterion used for the evaluation include both visual inspection, and false alarm and misdetection rates when groundtruth data are available. Lowe's algorithm has been widely cited and was found superior to most corner detection algorithms available [8] [7] [3].

5.1. Synthetic curves

Here we evaluate the performance of the two algorithms using the 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 0 and variance σ^2. The reconstructed test curves consist of perturbed sampled points. Figure 4 shows two synthetic curves adapted from those of Rosin [7] and Teh [9] respectively. Figure 5 shows the results from Lowe's algorithm (a and b) and our algorithm (c and d). Visually, both algorithms performed equally well on the two curves; but our algorithm outperformed Lowe's algorithm for both curves. Lowe's algorithm tends to detect local irregularity like small bumps or dips as corners, therefore yielding a higher false alarm rate.

Figure 4. Synthetic test curves adapted from Rosin (a) and Teh (b).

5.2. RADIUS data

A comparison was also carried out on 4 different RADIUS images. The goal here is to find the building corners

Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)
1063-6919/97 $10.00 © 1997 IEEE

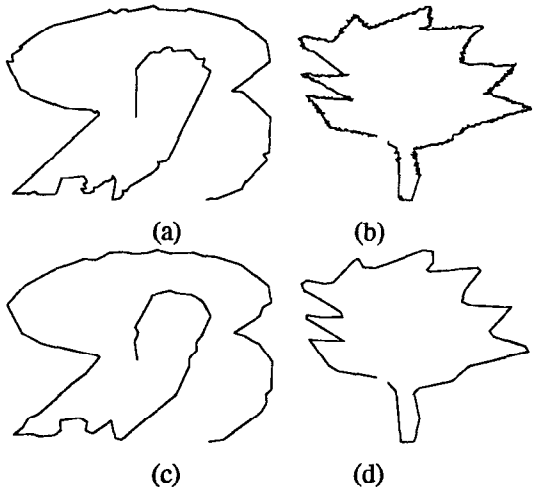

Figure 5. Results of polygonal approximation to the curves shown in figure 4 using Lowe's algorithm (a) and (b); and our algorithm (c) and (d), where window length=50 and angular threshold=25. Polygons are obtained by connecting the detected corners.

from the edge images of the buildings. Groundtruth building corners were obtained via the annotation procedure as described above. The criterion is therefore the misdetection and false alarm rates. Tables 1 and 2 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 much lower false alarm rates for all four images. Lowe's algorithm tends to make 2 to 7 times as many false alarm mistakes. This again demonstrates the superiority of our algorithm.

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

6. Discussion and Conclusions

In this paper, we have presented a statistical approach for detecting corners on digital arc sequences. A corner 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 corner if the angular orientations of the two lines that form the vertex are statistically significantly different. Performance evaluation study showed that the algorithm is robust and accurate for complex images. It has an average misdetection rate of 2.5% and false alarm rate of 2.2% for the complex RADIUS images. The study also revealed that our algorithm has consistently outperformed Lowe's technique on both synthetic and real data, with much lower false alarm rate. A major factor that contributes to the low false alarm rate of our algorithm is that we take the perturbation on the estimated line parameters into consideration, allowing us to treat the included angle as a random variable and statistically test its range. This represents a better model for corner detection than existing techniques, where the corner criteria like curvatures or tangent angles are treated as scalars rather than random variables, ignoring any perturbation they may be subject to and therefore leading to high false alarm rate for noisy images.

References

[1] J. G. Dunham. Optimal uniform piecewise linear approximation of planar curves. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6:67–75, 1986.

[2] R. M. Haralick. Propagating covariance in computer vision. *Proc. of 12th IAPR*, pages 493–498, 1994.

[3] T. Kadonaga and K. Abe. Comparison of methods for detecting corner points from digital curves. *Int. Workshop on Graphics Recognition*, 1995.

[4] D. G. Lowe. Three dimensional object recognition from single two dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

[5] T. Pavlidis. Waveform segmentation through functional approximation. *IEEE Tran. on Computers*, 22(7):689, 1973.

[6] T. Pavlidis and S. L. Horowitz. Segmentation of plane curves. *IEEE Transactions on Computers*, 23(8):860, 1974.

[7] P. L. Rosin and G. A. W. West. Nonparametric segmentation of curves into various representations. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(12):1140–1153, 1995.

[8] R. L. Rosin. Techniques for assessing polygonal approximations of curves. *7th British Machine Vision Conf., Edinburgh, 1996*, 1996.

[9] C. H. Teh and R. T. Chin. On the detection of dominant points in digital curves. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:859–872, 1989.

[10] K. Thornton and et al. Groundtruthing the radius modelboard imagery. *IUE Workshop*, 1995.

[11] J. A. Ventura and J. M. Chen. Segmentation of two-dimensional curve contours. *Pattern Recognition*, 25(10):1129–1140, 1992.

Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)
1063-6919/97 $10.00 © 1997 IEEE