A Bayesian Corner Detector

Xining Zhang, Robert M. Haralick, Visvanathan Ramesh and Anand S. Bedekar*
Intelligent Systems Laboratory
Department of Electrical Engineering, FT-10
University of Washington
Seattle, WA 98195

Ihsin Phillips
Department of Computer Science And Software Engineering
Seattle University
Seattle WA 98122

{xzhang,haralick,rameshv,anand,yun}@george.ee.washington.edu

Abstract

A corner is modelled as the intersection of two lines. A corner point is that point on an input digital arc whose *a posteriori* probability of being a corner is the maximum among all the points on the arc. The performance of the corner detector is characterized by its false alarm rate, misdetection rate, and the corner location error all as a function of the noise variance, the included corner angle, and the arc length. Theoretical expressions for the quantities compare well with experimental results.

1 Introduction

There are two primary groups of corner detection algorithms: one is based on detection directly from the underlying image[1-2], the other one is based on detection from arcs or curves[3-9] produced from previous low level image processing operations such as edge detection or line finding followed by thinning, linking and labeling. In addition, some researchers[10] have also explored corner detection based on combinations of these methods.

This paper presents a maximum *a posteriori* (MAP) probability corner detection method. For a given arc segment, the corner is estimated to be that point whose *a posteriori* probability of being a corner is the maximum among all the points on the arc segment.

We model an ideal corner as the intersection point of two straight lines. Our mathematical formulation of the corner detection incorporates the prior distributions for corner model parameters, such as the parameters of the lines forming the corner and the index of the corner point along the arc segment.

The detection procedure involves sliding a context window of specified length over the given sequence of pixels forming the arc segment, and doing a two-line segment corner detection within each window. The context window length is chosen so that there is at most one corner within the context window.

*Funding from DARPA contract 92-F1428000-000 is gratefully acknowledged.

2 Motivation and Theory

A corner is a discontinuity of the curvature of a curve and the location of the discontinuity can be approximated by the intersection point of two straight lines in its local neighborhood.

2.1 Corner Model and Its Detector

Given an observed sequence of ordered points arising from two line segments, the last observed point arising from the first estimated line segment is what we want to detect as the corner. The problem is to decide which of the points in the observed sequence has the maximum *a posteriori* probability of being the "last" point from the first line segment. The following is the formalized problem statement:

Problem Statement

Given: an observed sequence of points from an arc segment, $\hat{S} = \{\begin{pmatrix}\hat{r}_i \\ \hat{c}_i\end{pmatrix} \mid i = 1, ..., I; (\hat{r}_i, \hat{c}_i) \in Z_R \times Z_C\}$, where $Z_R \times Z_C$ is the image domain, I is the number of points and $(\hat{r}_i, \hat{c}_i), i = 1, ..., I$ are the results of random perturbations on the points $(r_i, c_i), i = 1, ..., I$ constrained by

$$\begin{aligned} r_i cos\theta_1 + c_i sin\theta_1 - \rho_1 &= 0, i = 1, ..., k; \\ r_i cos\theta_2 + c_i sin\theta_2 - \rho_2 &= 0, i = k+1, ..., I, \end{aligned}$$

where $\theta_j, \rho_j; j = 1, 2$ are line orientation and location parameters for the two line segments. and k is the index of the true corner position (r_k, c_k). Assume perturbations to be independently introduced on each sample point with Gaussian distributed noise in the direction perpendicular to the line segment. Perturbations on the two line segments can be expressed by

$$\begin{aligned} \hat{r}_i &= r_i + \eta_i cos\theta_1; \hat{c}_i = c_i + \eta_i sin\theta_1; i = 1, ..., k; \\ \hat{r}_i &= r_i + \eta_i cos\theta_2; \hat{c}_i = c_i + \eta_i sin\theta_2, ; i = k+1, ..., I. \end{aligned}$$

where $\eta_i \sim N(0, \sigma^2)$.

Find: the estimated corner $(\hat{r}_{k^*}, \hat{c}_{k^*}), 2 \leq k^* \leq I - 1$, along the arc $\hat{S}$ and the estimates of two line parameters, (θ_1^*, ρ_1^*) and (θ_2^*, ρ_2^*) so that $P(k, \theta_1, \rho_1, \theta_2, \rho_2 \mid \hat{S}, \sigma, I)$ is maximized.

Solution: By Bayes' formula $P(k,\theta_1,\rho_1,\theta_2,\rho_2 \mid \hat{S},\sigma,I)$ can be written as $P(\hat{S} \mid k,\theta_1,\rho_1,\theta_2,\rho_2,\sigma,I)P(k,\theta_1,\rho_1,\theta_2,\rho_2 \mid \sigma,I)$.

The first term is the likelihood of observing the given sequence of points $\hat{S}$, given the parameters of the two lines forming the corner, the noise standard deviation σ, I and the index k of the true corner. The model says that the observed sequence $\hat{S}$ can be separated into two sub-sequences, or sub segments, $\hat{S}_1$ and $\hat{S}_2$, where, $\hat{S}_1 = \{(\begin{smallmatrix}\hat{r}_i\\ \hat{c}_i\end{smallmatrix}) \mid i = 1,..,k\}$ and $\hat{S}_2 = \{(\begin{smallmatrix}\hat{r}_i\\ \hat{c}_i\end{smallmatrix}) \mid i = k+1,...,I\}$. Since perturbations on the first line (θ_1,ρ_1) are independent from those on the second line (θ_2,ρ_2), the likelihood of the observed $\hat{S}$ given two lines (θ_1,ρ_1), (θ_2,ρ_2) can be written as

$$P(\hat{S} \mid k,\theta_1,\rho_1,\theta_2,\rho_2,\sigma,I) = P(\hat{S}_1 \mid k,\theta_1,\rho_1,\sigma,I)P(\hat{S}_2 \mid k,\theta_2,\rho_2,\sigma,I).$$

The conditional probability of observing the first sub segment given the true line parameters is given by

$$\begin{aligned} &P(\hat{S}_1 \mid k,\theta_1,\rho_1,\sigma,I) \\ &= P(\hat{S}_1 \mid k,\theta_1,\rho_1,\sigma) \\ &= P((\hat{r}_1,\hat{c}_1)',...,(\hat{r}_k,\hat{c}_k)' \mid \theta_1,\rho_1,\sigma) \\ &= \prod_{i=1}^{k} P((\hat{r}_i,\hat{c}_i)' \mid \theta_1,\rho_1,\sigma) \\ &= (\frac{1}{\sqrt{2\pi}\sigma})^k \prod_{i=1}^{k} e^{-\frac{1}{2\sigma^2}(\hat{r}_i cos\theta_1+\hat{c}_i sin\theta_1-\rho_1)^2}. \end{aligned}$$

Similarly, the conditional probability of observing the second sub segment $\hat{S}_2$ can be computed by

$$P(\hat{S}_2 \mid k,\theta_2,\rho_2,\sigma,I) = (\frac{1}{\sqrt{2\pi}\sigma})^{I-k} \prod_{i=k+1}^{I} e^{-\frac{1}{2\sigma^2}(\hat{r}_i cos\theta_2+\hat{c}_i sin\theta_2-\rho_2)^2}.$$

The index k and parameters (θ_1,ρ_1), (θ_2,ρ_2) are independent of σ, hence

$$P(k,\theta_1,\rho_1,\theta_2,\rho_2 \mid \sigma,I) \;=\; P(k,\theta_1,\rho_1,\theta_2,\rho_2 \mid I).$$

Further, the index k is independent from the line parameters (θ_1,ρ_1) (θ_2,ρ_2), and these line parameters are independent of the number of points I. So

$$\begin{aligned} &P(k,\theta_1,\rho_1,\theta_2,\rho_2 \mid I) \\ &= P(\theta_2,\rho_2,\theta_1,\rho_1 \mid I)P(k \mid I) \\ &= P(\theta_2,\rho_2,\theta_1,\rho_1)P(k \mid I) \\ &= P(\theta_2 \mid \rho_2,\theta_1,\rho_1)P(\rho_2 \mid \theta_1,\rho_1)P(\rho_1 \mid \theta_1)\cdot \\ &\quad P(\theta_1)P(k \mid I) \\ &= P(\theta_2 \mid \theta_1)P(\rho_2)P(\rho_1 \mid \theta_1)P(\theta_1)P(k \mid I). \end{aligned}$$

The index k, i. e. the index of the last point arising from the first line, is assumed to be uniformly distributed between the second point and the second-from-last point, i. e. θ_1 is assumed to be uniformly distributed in $[0,2\pi]$, i. e. $P(\theta_1) = 1/2\pi$.

The conditional probability distribution $P(\rho_1 \mid \theta_1)$ is a probability density of the distance ρ of the line from the origin, given θ_1, which is the orientation of the vector normal to the line.

We assume the probability distribution of ρ_1 given θ_1 is constant and equal to $1/Z$, where Z is the larger of the number of rows or columns in the image. We assume ρ_2 to be uniformly distributed in $[0 \le \rho_2 < Z]$ and has zero probability in the region of $[Z \le \rho_2 \le \sqrt{2}Z]$

The conditional probability distribution $P(\theta_2 \mid \theta_1)$ is assumed to be determined just by the angle included between the two lines. $P(\theta_2 \mid \theta_1) = P(\mid \theta_2 - \theta_1 \mid)$.

Let $\theta_{12} = \mid \theta_2 - \theta_1 \mid \in [0,\pi]$. θ_{12} is called the included corner angle. It is assumed that there is a higher probability that the included angle is close to a right angle. This assumption is consistent with some practical applications such as roof corner detection of buildings in aerial images[12-13]. We assume the probability distribution of θ_{12} to be

$$P(\theta_{12}) = K_1 e^{K_2 \sin(\theta_{12})},$$

where K_1 and K_2 are two constants.[1] K_2 can be estimated from the empirical distribution of θ_{12} by $\hat{K}_2 = 1/\hat{\sigma}^2_{\theta_{12}}$, and K_1 can be estimated by

$$\hat{K}_1 = \frac{1}{\int_0^{\pi} e^{\hat{K}_2 \sin(\theta_{12})} d\theta_{12}},$$

where $\hat{\sigma}^2_{\theta_{12}}$ is the estimated variance of the empirical distribution of θ_{12}.

Taking logarithms, the problem becomes that of finding the $(k^*,\theta_1^*,\rho_1^*,\theta_2^*,\rho_2^*)$ that maximizes

$$\begin{aligned} K &- \frac{1}{2\sigma^2}\sum_{i=1}^{k}(\hat{r}_i cos\theta_1 + \hat{c}_i sin\theta_1 - \rho_1)^2 - \\ &\frac{1}{2\sigma^2}\sum_{i=k+1}^{I}(\hat{r}_i cos\theta_2 + \hat{c}_i sin\theta_2 - \rho_2)^2 + \\ &\qquad + K_2 \sin(\mid \theta_2 - \theta_1 \mid), \end{aligned}$$

where

$$K = \log K_1 - \log(2\pi) - 2\log Z - \log(I-2) - I\log(\sqrt{2\pi}\sigma).$$

The above problem is a nonlinear optimization problem. We use a two step procedure to find the solution. In the first step, we use maximum likelihood estimation to quickly find a good initial estimate and in the second step, we make use of a gradient search scheme to find the solution to the MAP problem.

[1] This distribution is nothing but a truncated form of the Von Mises distribution with mean= $\pi/2$.

2.2 Application to Multi Linear Segment Models

The procedure begins by examining the first I pixels in the chain. If a corner is detected, the corner detector puts the next context window to start at the point following the detected corner. If no corner is detected, the window is moved along the pixel chain in a fixed step size (usually one pixel) and the corner detector is reapplied.

For any context window, there is always one whose *a posteriori* probability of being a corner is the maximum among all the points in the window. However, there may actually be no true corner located in the context window. In this case, the point with the MAP probability of being a corner should not be claimed as a detected corner point. Therefore, we need a way of determining whether the MAP probability of a point being a corner is high enough so that the detector should label the point as a corner.

We first concentrate on the likelihood part of the detector. The maximization of the likelihood of observing the given sequence from the estimated corner model is equal to the minimization of

$$X_1 = \frac{1}{\sigma^2}\sum_{i=1}^{k}(\hat{r}_i \cos\theta_1 + \hat{c}_i \sin\theta_1 - \rho_1)^2 + \frac{1}{\sigma^2}\sum_{i=k+1}^{I}(\hat{r}_i \cos\theta_2 + \hat{c}_i \sin\theta_2 - \rho_2)^2 \quad (1)$$

When there is no corner in the window, $\theta_1 = \theta_2$ X_1 would be a χ^2 distributed random variable with I degrees of freedoms.

We choose a confidence level α_T and set a threshold T_p so that

$$Prob(X_1 < T_p) = \alpha_T.$$

This is the probability that a χ^2 random variable with I degrees of freedom is less than T_p. In reality, we have two estimates for the angles of the lines $\hat{\theta}_1$ and $\hat{\theta}_2$ and we compute the quantity:

$$X_2 = \min\{\frac{1}{\sigma^2}\sum_{i=1}^{I}(\hat{r}_i \cos\hat{\theta}_1 + \hat{c}_i \sin\hat{\theta}_1 - \rho_1)^2, \frac{1}{\sigma^2}\sum_{i=1}^{I}(\hat{r}_i \cos\hat{\theta}_2 + \hat{c}_i \sin\hat{\theta}_2 - \rho_2)^2\}, \quad (2)$$

and then compare X_2 with the determined threshold T_p. If X_2 is larger than T_p, the estimated break point is claimed as a detected corner. Otherwise, the estimated breaking point is not claimed as a corner.

If, according to the above criterion, no corner is claimed, the detector is moved along the given arc by a defined step (usually one pixel). If a corner is detected, the detector is moved to the next window starting at the pixel next to the detected corner. This procedure is repeated until the tail of the detector window reaches the last point of the given arc.

3 Location Error

The squared location error d^2 is the squared distance between the detected corner position and the true corner position,

$$d^2 = (\hat{r}_{k^*} - r^o)^2 + (\hat{c}_{k^*} - c^o)^2, \quad (3)$$

where k^* is the index of the estimated corner position (r^*, c^*), (r^o, c^o) is the true corner position, i. e. the intersection of the two lines forming the corner. The variance of the squared distance can be proved to be

$$V[d^2 \mid \Phi] = E\{[(r_{k^*} - r^o)^2 + (c_{k^*} - c^o)^2]^2 \mid \Phi\} - \{E[(r_{k^*} - r^o)^2 \mid \Phi] + E[(c_{k^*} - c^o)^2 \mid \Phi]\}^2 + 2\sigma^4,$$

where

$$E[(r_{k^*} - r^o)^2 \mid \Phi] = \sum_{i_{1^*}=2}^{I-1}(r_{k^*} - r^o)^2 P(k^*; \Phi)$$

$$E[(c_{k^*} - c^o)^2 \mid \Phi] = \sum_{i_{1^*}=2}^{I-1}(c_{k^*} - c^o)^2 P(k^*; \Phi)$$

and

$$E[\{(r_{k^*} - r^o)^2 + (c_{k^*} - c^o)^2\}^2 \mid \Phi] = \sum_{i_{1^*}=2}^{I-1}[(r_{k^*} - r^o)^2 + (c_{k^*} - c^o)^2]^2 P(k^*; \Phi)$$

where $P(k^*; \Phi)$ is the notation used to denote the probability that the observed sequence $\hat{S}$ is such that the corner estimated from it has index k^*, given that the parameters of the underlying lines are Θ and the noise standard deviation is σ.

In the experiments for examining the location error as a function of the noise standard deviation σ, for each value of σ, many trials are performed with the same value of the included corner angle. For each trial, an pair of line segments is generated and is sampled, and then the samples are perturbed with Gaussian noise of variance σ^2. For each of the N trials there is a different observed sequence $\hat{S}(n)$, $n = 1, \ldots, N$, and the observed points in $\hat{S}(n)$ are substituted in the above expression to obtain an estimate of $E(d^2|\Phi)$. Thus an estimate of $E(d^2|\Phi)$ is obtained from each trial, and these are then averaged to obtain a better estimate of $E(d^2|\Phi)$.

4 Experimental Protocol and Results

The experiments utilize synthetically generated two-line-segment sequences. The input parameters to the corner detector are the context window length cwl, the estimated standard deviation of the noise σ, and the confidence coefficient α_{T_p}.

4.1 Two Line Segment Arc Generation

A two line segment arc can be generated in three steps: (1) Specify the starting point (r_1, c_1), the first line length L_1, the second line length L_2, the first line angle ϕ_1 and the included corner angle θ_{12}, where ϕ_1 is the counterclockwise angle between the first line and the row axis. (2) Generate a sequence of samples $< m_i; i = 1, .., I >$, where each m_i; $i = 1, ..., I$ is an independent random sample coming from a Gaussian distributed random variable with zero mean and a standard deviation σ. (3) Obtain a perturbed sequence of the arc segment $< (r_i + m_i \cos\theta_1, c_i + m_i \sin\theta_1); i = 1, \ldots, i^t >$, and $< (r_i + m_i \cos\theta_2, c_i + m_i \sin\theta_2); i = i^t + 1, \ldots, I >$. Thus, a perturbed two line segment arc $\hat{S} =< (\hat{r}_i, \hat{c}_i); i = 1, ..., I >$ is generated.

4.2 Location Error Measurement

For measuring the location error, we utilize synthetically generated two-line-segment sequences, and obtain the distance between the true corner and the detected corner. In this experiment, the context window length is equal to the length of the sequence, σ is systematically set up, and α_T is not used.

Location error vs noise standard deviation

Let $\theta_{12} = 90°$, $L_1 = L_2 = 50$ units. For each $\sigma \in \{0.0, 0.2, 0.4, ..., 5.0\}$ and for all of $\theta_1 \in \{0°, 1°, ..., 359°\}$, generate 10 sequences of two-line-segment arcs. There are 360*10 runs, defined as N_{run}, for each σ. For each sequence, apply the corner detector, and obtain the squared location error by

$$d_p^{2(n)} = (\hat{r}_{i_1}^{(n)} - r^{t(n)})^2 + (\hat{c}_{i_1}^{(n)} - c^{t(n)})^2,$$

where $(\hat{r}_{i_1}^{(n)}, \hat{c}_{i_1}^{(n)}); n = 1, ..., N_{run}$ is the estimated corner and $(r^{t(n)}, c^{t(n)}); n = 1, ..., N_{run}$ is the related true corner. Obtain the root-mean-square error by

$$\bar{d}_p = (\frac{1}{N_{run}} \sum_{n=1}^{N_{run}} d_p^{2(n)})^{\frac{1}{2}},$$

and its related variance by

$$var(d_p) = (\frac{1}{(N_{run} - 1)} \sum_{n=1}^{N_{run}} (d_p^{2(n)} - \bar{d}_p^2)^2)^{\frac{1}{2}}.$$

Figure ?? shows the root-mean-square location error and the root-mean-square variance of the location error versus the noise standard deviation, by theory and experiment. It indicates that the error linearly increases as the noise increases and the variance of the error quadratically increases as the noise increases.

Location error vs the included corner angle

This experiment is the same as above except that θ_{12} is chosen from the set $\{10°, 20°, ..., 170°\}$ and $\sigma = 1.0$

Figure ?? illustrates the root-mean-square location error vs θ_{12}, and indicates that the detection has the tendency of having smaller error for 90° included corner angle and larger error for included corner angles away 90°. In addition, the rather flat region around

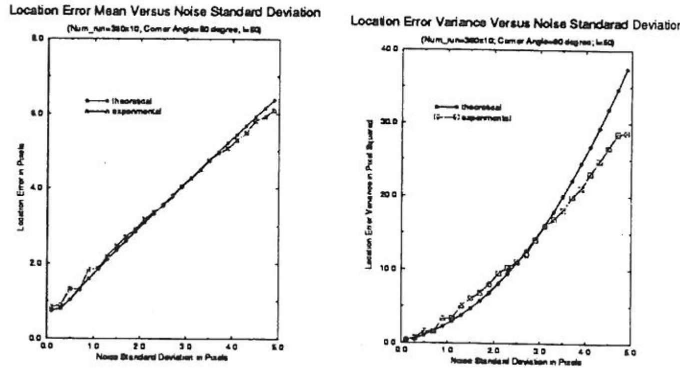

Figure 1: Location error and its Variance Versus σ

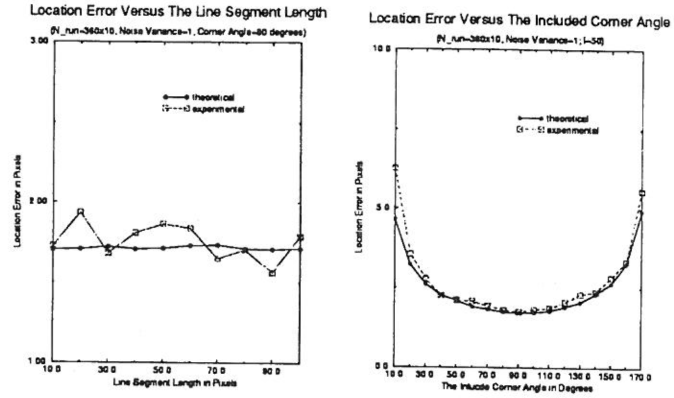

Figure 2: Location Error vs Included Corner Angles And Arc Length

90° corner angle indicates that the algorithm is more stable over a large range of corner angles.

Location error vs the arc sequence length

This experiment is the same as that in the first case but at this time the arc length is varied from 10 to 100 by a step of 10 pixels and the two line segment lengths are kept the same, ie. $L_1 = L_2$.

Figure ?? is the root-mean-square location error versus the arc length. The result indicates that the algorithm is stable with different arc lengths.

4.3 Performance Measurement

Once the algorithm has been designed, it's performance should be characterized [14]. In this experiment, we test the performance of the detector by plotting its false alarm rate and misdetection rate versus the context window length *cwl*, the included corner angle θ_{12} and the distance threshold d_0 which is a special parameter used during performance test.

Here, *cwl* is chosen smaller than the sequence length, σ is systematically set up, and $\alpha_T = 0.9$.

The false alarm rate is defined as the probability of a true noncorners being detected as a corner and the misdetection rate is defined as the probability of a true corner not being detected as a corner. Define a circle of radius d_0, called the distance threshold, centered at a true corner. If no point exists within this circle,

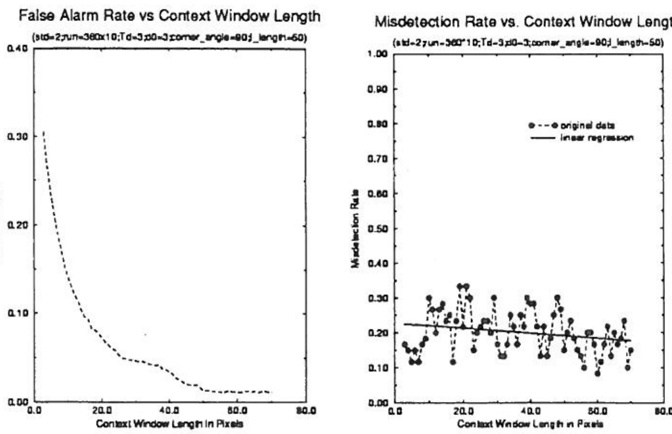

Figure 3: False Alarm and Misdetection versus Context Window Length

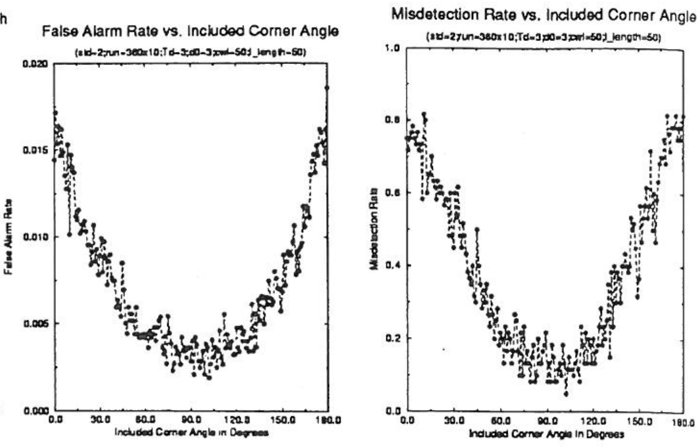

Figure 4: False Alarm and Misdetection Versus Included Angle

a misdetection happens. If the detected corner does not fall into any region centered by a true corner with the given radius d_0, this detection is claimed as a false alarm.

1. **False alarm rate and misdetection rate versus the context window length.**

   Let $\theta_{12} = 90°$, $L_1 = L_2 = 50$ units, $T_d = 3$. For each $cwl \in \{3, 4, ..., 70\}$, where $cwl < 2 * 50$. and all of $\theta_1 \in \{0°, 1°, ..., 359°\}$, generate 10 sequences of two-line-segment arcs. For each context window length, there is $N_{run} = 360 * 10$ runs. For each generated curve, detect corners. The false alarm and misdetection rates are shown in Figure ??.

2. **False alarm rate and misdetection rate versus the included corner angle.**
   This experiment is the same as above except that the included corner angle θ_{12} at this time is varied from 1° to 179° by a step size of 1° and the context window length $cwl = 2 * 50$ unit.

   The results are shown in Figure ??.

References

[1] Zuniga, O. A., and R. M. Haralick, "Corner detection using the facet model", *Proc. of IEEE CVPR*, pp.30-37, 1983.

[2] Deriche, R., and G. Giraudon, "Accurate corner detection: an analytical study", *Proc. of IEEE ICCV*, 1990, pp.66-70.

[3] Pavlidis, T."Algorithms for shape analysis of contours and waveforms", *IEEE Trans. PAMI-2*, pp.301-312, 1980.

[4] Dunham, J. G., "Optimum uniform piecewise linear approximation of planar curves", *IEEE Trans. PAMI-8*, pp.67-75, 1986.

[5] Ventura, J. A., and J.-M. Chen, "Segmentation of two-dimensional curve contours", *Pattern Recognition,* Vol 25. No.10, pp.1129-1140, 1992

[6] Phillips, T.Y., and A. Rosenfeld, "An ISODATA algorithm for straight line fitting", *Pattern Recognition Letters,* Vol. 7, 1988, pp. 291–297.

[7] Zhang, X., and R. M. Haralick, "Bayesian corner detection", *Proc. of British Machine Vision Conference* BMVC'93, University of Surrey, Guildford. U.K. 21-23 September 1993.

[8] Hemminger, T. L., and C. A. Pomalaza-Raez, "Polygonal representation: a maximum likelihood approach", *Comput. Vision Graphics Image Process,* **52** pp.239-247, 1990.

[9] Teh, C.-H., and R.T. Chin, "A scale-independent dominant point detection algorithm", in *Proceedings of CVPR,* Ann Arbor, Michigan, 1988, pp. 229–234.

[10] Xie, X., R. Sudhakar, and H. Zhuang, "Corner detection by a cost minimization approach", *Pattern Recognition,* Vol. 26, No. 8. pp.1235-1243, 1993.

[11] Ramesh, V. and R.M.Haralick, "Performance characterization of Edge operators," *SPIE Machine Vision and Robotics Conference,* Orlando, FL., April 92.

[12] Haralick, R.M., "Edge and region analysis for digital image data," *Computer Graphics and Image Processing,* Vol.12, 1980, pp. 60-73.

[13] F.J.Canny, "Finding edges and lines in images," *Tech.Rep. 720, MIT AI Lab,* June 1983.

[14] Ramesh, V. *et al,* "Automatic Tuning Parameter Selection for Feature Extraction Sequence," *Proc. of CVPR,* Seattle, Washington, pp.672-677, June 1994.