Corner Detection via Topographic Analysis of Vector Potential

Bin Luo †, ‡ A. D. J. Cross † E. R. Hancock †
†Department of Computer Science,
University of York,York YO1 5DD, UK.
‡Anhui University, P.R. China
[luo|adjc|erh]@minster.cs.york.ac.uk

Abstract

This paper describes how corner detection can be realised using a new feature representation that has recently been successfully exploited for edge and symmetry detection. The feature representation based on an magneto-static analogy. The idea is to compute a vector potential by appealing to an analogy in which the Canny edge-map is regarded as an elementary current density residing on the image plane. In our previous work we demonstrated that edges are the local maxima of the vector potential while points of symmetry correspond to the local minimum. In this paper we demonstrate that corners are located at the saddle points of the magnitude of the vector potential. These points correspond to the intersections of saddle-ridge and saddle-valley structures, i.e. to junctions of the edge and symmetry lines. We describe a template-based method for locating the saddle-points. This involves performing a non-minimum suppression test in the direction of the vector potential and a non-maximum suppression test in the orthogonal direction. Experimental results of both synthetic and real images are given. Comparisons of the method in different scales and with the SUSAN corner detector are also given.

1 Introduction

Corners are important dominant points in digital images. In many computer vision tasks, such as image registration, image matching [4], object recognition [15, 10] and motion analysis[7], accurate corner detection is essential. Broadly speaking, there are two corner-detection strategies adopted in literature. The first of these is based on the analysis of pre-segmented contours, while the second is based on the differential analysis of the raw gray-scale image. However, in both cases it is the rate of change of contour angle that is used to characterise corner features.

In the case of boundary-based corner detection. from presegmented contours there are three processing steps. Firstly, the image is pre-segmented. Secondly, boundaries of the object in the image are extracted and chain coded. Finally, algorithms are developed to detect corners along the boundaries. According to the contour-based approach, corners are defined as the intersection points or junction points between straight line segments [25]. Several authors have used chain codes to provide a digital characterisation of corners

BMVC 1998 doi:10.5244/C.12.57

[8, 2, 13, 20, 1]. A review is provided by Liu [14]. The main weakness of the contour-based method of corner detection is the availability of a reliable image segmentation.

Corners in gray-scale images are characterised by using the second-derivatives of the image luminance function [23]. Although this method does not require pre-segmented image contours, it is sensitive to the noise amplification effects of the second-derivative operators. Gray-scale corner detection algorithms can be divided into two groups, Template-based corner detector [19, 16] detect the similarity between a given template of a specific angle for each image sub-window. Because multiple orientation templates are used, the technique is computationally expensive. Gradient-based corner detector [12, 21, 6, 18, 24], on the other hand, relies on measuring the curvature of an edge that passes through a neighbourhood. The strength of the corner response depends on both the edge strength and the rate of change of edge direction. Gradient-based corner detection techniques are more likely to respond to noise than their contour-based counterparts, and often perform quite poorly.

Our aim in this paper is to present a new corner detection method which exploits some of the best features of the template based and the curvature-based methods. In particular, we aim to detect corners via direct topographic analysis. Unlike Haralick's topographic primal sketch[11] which is based on the topography of gray-scale features, we analyse the topographic structure of the vector potential representation. Corners are located where saddle points occur. We commence from an image representation which has already been shown to provide a convenient topographic representation for edge and symmetry features [5]. The representation is based on a magneto-static analogy in which the raw Canny edge-map is responsible for generating a current density which resides on the image plane. The key idea is to compute the associated magnetic potential at various heights above the image plane. Local maxima (i.e. ridges) of the magnitude of the vector potential correspond to edges while the local minima (i.e. ravines) are symmetry lines. According to this topographic picture, corners are saddle-points of the magnitude of the vector potential.

With the topographic representation to hand, the main practical problem that confronts us is the localisation of the saddle-points. This is more difficult than localising ridges and ravines since we are concerned with identifying point features rather than contour features. In the case of ridges and ravines we can exploit constraints on compatible continuity or directionality. In the case of saddle points the constraints are more subtle, since we are seeking locations which are consistent with being the junctions between saddle-ridges and saddle-valleys. Based on this observation we develop topographic tests for the consistent saddle-structure in the vector potential. This is effectively a template based method. We search for consistent valley structure in the direction of the vector potential and consistent ridge structure in the orthogonal direction. In other words, the directional template characterise local saddle-structure as the intersection of ridge (i.e. edge) and ravine (i.e. symmetry) structures. Moreover, computation is simplified since the template is fixed to be in the direction of the vector-potential. This is a threshold-free operation. The final corner selection is based on thresholding the local change in orientation of the vector potential.

2 Image representation using vector potential

In this section, we review the feature-representation recently reported by Cross and Hancock [5]. The starting point is to compute the Canny edge map[3]. Accordingly, we commence by convolving the raw image I with a Gaussian kernel of width σ. The kernel takes the following form

$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2}\exp\left[-\frac{x^2+y^2}{2\sigma^2}\right] \tag{1}$$

With the filtered image to hand, the Canny edge map is recovered by computing the gradient $\mathrm{E} = \nabla G_\sigma * I$

In order to compute a vector field representation of the edge-map, we will need to introduce an auxiliary z dimension to the original $x-y$ co-ordinate system of the plane image. In this augmented co-ordinate system, the components of the edge-map are confined to the image plane. In other words, the edge-vector at the point $(x,y,0)$ on the input image plane is given by

$$\mathrm{E}(x,y,0) = \begin{pmatrix} \frac{\partial G_\sigma * I(x,y)}{\partial x} \\ \frac{\partial G_\sigma * I(x,y)}{\partial y} \\ 0 \end{pmatrix} \tag{2}$$

In order to pursue our magneto-static analogy we would like to interpret the raw edge responses as elementary currents which flow around the boundaries and give rise to a vector potential. The elementary current-vector at the point $(x,y,0)$ on the input image plane is defined to be

$$\mathrm{A}(x,y,z) = \mu\int_{V'}\frac{\hat{z}\wedge\nabla G_\sigma * I(x,y)}{|\mathrm{r}-\mathrm{r}'|}dV' \tag{3}$$

3 Differential Operators

In order to develop the appropriate differential operators for feature characterisation from the vector-potential we have appealed to the geometry of the associated magnetic field. According to magneto-statics the magnetic field is the curl of the vector potential. It is important to stress that because it is less computationally tractable than the vector-potential, the magnetic field is never used directly in our image representation. The role of the magnetic field is to provide an auxiliary representation. The geometry of the magnetic field allows us to understand the differential structure of the vector-potential.

According to our representation of image structure, symmetry lines follow the local minima of the vector-potential. In other words, they connect image points where there is strong cancellation edge tangent vectors associated with symmetrically placed object boundaries. By contrast, edge contours follow the local maxima of the vector potential. According to our representation, the edge lines connect points where there is strong directional re-enforcement between edge tangent-vectors. Symmetry lines can be interpreted as locations where the magnetic field is perpendicular to the sampling image plane. Edges are locations where field lines are tangential to the relevant sampling plane. When viewed from perspective of the differential structure of the vector potential, symmetry lines are locations where the component of the curl in the image plane vanishes, i.e. $\hat{z}\wedge\nabla\wedge\mathrm{A}(x,y,z) = 0$; edges are locations where the transverse component of the divergence vanishes, i.e. $\nabla.(\hat{z}\wedge\mathrm{A}(x,y,z)) = 0$.

Corners, or points of locally maximum boundary curvature, can be viewed as edge-locations where there is a local symmetry axis associated with a rapid change in boundary direction. When viewed from the perspective of our image representation, corners therefore correspond to locations where both the edge and symmetry conditions are simultaneously satisfied. From a topographic viewpoint, corners are located where boundary lines and symmetry lines meet. In other words, we are interested in locating points where there is a local maximum of the magnitude of the vector potential in one direction and a local minimum in the orthogonal direction. As a result corner detection can be treated as saddle point detection.

4 Topographic Representation

In the previous section, we established that corners are saddle-points in the magnitude of the vector-potential. We therefore focus on the analysis of the scalar quantity

$$g(x,y)_z = |\mathbb{A}(x,y,z)| \tag{4}$$

The topographic structure of the vector potential can be characterised up to second-order using the gradient ∇g and Hessian matrix $\mathcal{H}_g$.

The Gaussian curvature H and mean curvature K can be computed as follows

$$H = \frac{det(\nabla\nabla^T g)}{(1+\|\nabla g\|^2)^2} \tag{5}$$

$$K = \frac{1}{2}\nabla \cdot \left(\frac{\nabla g}{\sqrt{1+\|\nabla g\|^2}}\right) \tag{6}$$

where ∇ is the 2-D gradient operator, $\nabla\nabla^T$ is the Hessian matrix operator and $(\nabla\cdot)$ is the divergence operator of vector calculus. Therefore, the Gaussian curvature and the mean curvature can computed directly using first and second partial derivatives.

Class	Symbol	K	H	Region-type
Dome	D	-	+	Elliptic
Ridge	R	-	0	Parabolic
Saddle ridge	SR	-	-	Hyperbolic
Plane	P	0	0	Hyperbolic
Saddle-point	S	0	-	Hyperbolic
Cup	C	+	+	Elliptic
Valley	V	+	0	Parabolic
Saddle-valley	SV	+	-	Hyperbolic

Table 1: Curvature classes

The signs and zeros of the mean and Gaussian curvatures can be used to categorise the local surface geometry into a number of distinct topographic classes. These classes are summarised in Table 1. In this paper, we are interested in the saddle-structures which

are labelled as hyperbolic features in the table. These features are characterised by the condition $H < 0$. In particular we are interested in points that are consistent with being the intersections of edge and symmetry lines, i.e. in the intersections of saddle-ridges and saddle valleys. The joint condition for the intersections is $K \neq 0 \ \wedge \ H < 0$.

By searching for the intersection of consistent saddle ridges, we overcome some of the problems of localising saddle-points, for which $K = 0$ and $H < 0$. This can prove difficult since there are no constraints from the directionality of the desired feature. In particular, we mitigate this difficulty and realise the corner localisation process using templates to search for the junctions between symmetry lines and edge-lines.

5 Implementation

Based on the results presented in the previous section, we make the following observations concerning the topographic structure of the vector potential in the proximity of corners

(1)There is a local minima of the magnitude of the vector potential in the direction of the vector-potential

(2)There is a local maxima of the magnitude of the vector potential in the orthogonal direction.

(3)At the locations of corners, the magnitude of vector potential along both the contour and its orthogonal direction changes rapidly.

(4)At the locations of corners, the phase of the vector potential changes rapidly along the directions mentioned above.

With the first two observations we search for saddle-points that are consistent when viewed from a finite support neighbourhood. In practice we localise consistent topographic structure using a simplified form of template convolution. Our template tests for orthogonal maxima and minima using non-maximum suppression and non-minimum suppression tests. The saddle points are corner candidates.

Because of image noise and other imperfections, the points detected by our saddle-template are not always the locations of true corners in the image. To overcome this problem we can appeal to the directional consistency of the derivatives of the vector potential to refine the corner estimates.

Th meet this goal, we define a corner "strength" measure. It measures the directional variations of the vector potential along the contour and orthogonal to its direction. Let $\mathbf{A}$ be the vector potential, $\nabla_\theta A$ is the gradient of the magnitude of the vector potential in the contour direction θ. $\nabla_{\theta\perp} A$ is gradient of the scalar potential in orthogonal direction. The corner strength is defined as,

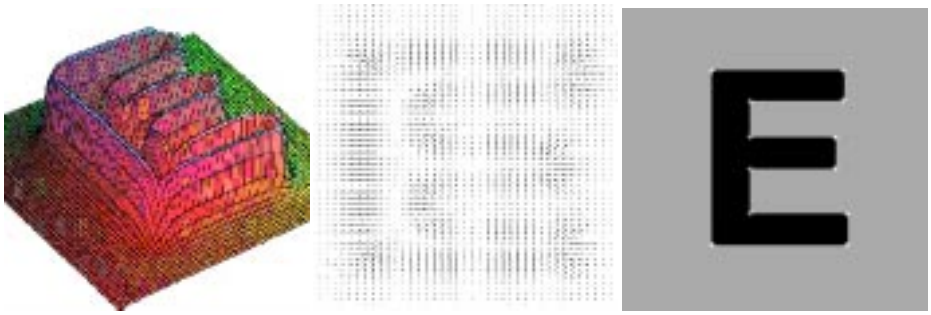
$$C = |\nabla_\theta A| * |\nabla_{\theta\perp} A| \tag{7}$$

Corners are selected by thresholding the corner strength.

6 Experiments

In this section, we provide some experimental evaluation of the corner detection algorithm. The experiment work is divided into two parts. We commence with some examples on binary imagery to illustrate some of the properties of the representation. Next we furnish real world examples.

To illustrate the properties of our vector potential representation and corner detection algorithm, we use an simple binary image of "E". Figure 1(a) is the magnitude of the vector potential for the binary image, Figure 1(b) shows the detected corners. For this simple image, the results are all correct. Figures 1(c) is the direction of the vector potential. The magnitude of the vector potential is displayed as a height in Figure 1(a) to emphasise topographic structure. Here the saddle-structure associated with the corners is clear. The ridge and ravine structure of the edge and symmetry lines is also evident. In Figure 1(c) we display the vectorial representation of $\mathbf{A}(x, y, 0)$. The main feature to note from this figure is that the direction of the vector potential changes rapidly at the corner locations.

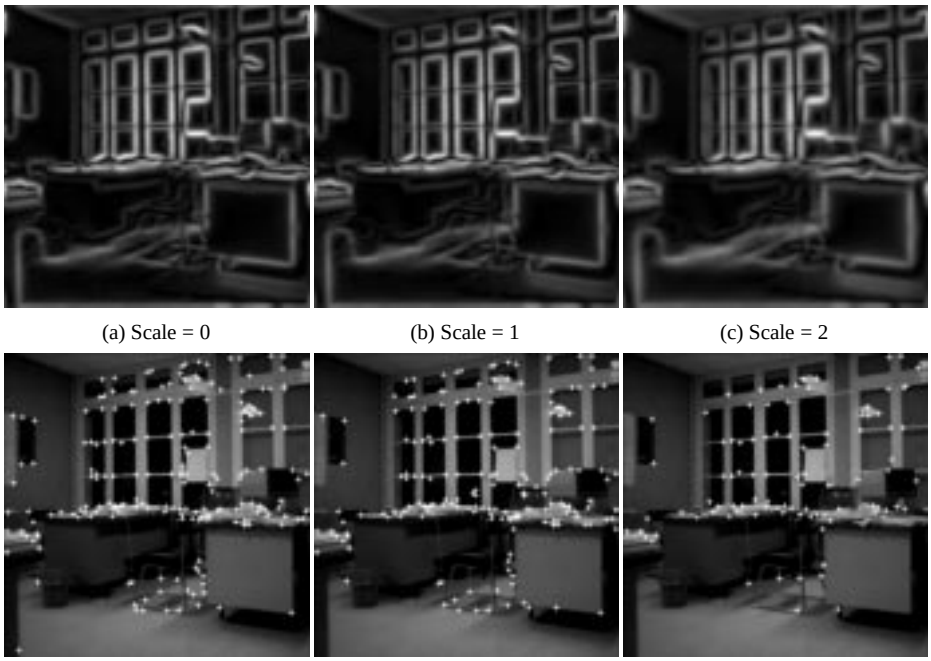


(a) Magnitude (b) Direction (c) Detected corners

Figure 1: Topographic representation and corner detection

As explained earlier, our image representation has the scale-space property. In the second experiment, we use a real image to test our corner detection algorithm and show the results at different scales. The image used in this study is the INRIA office scene. Figure 2 shows the results of corner detection at number of different scales. The upper-row of the figure shows the magnitude of the vector potential, while the lower-row shows the detected corners superimposed on the original image. As we move from the left column to the right column, the sampling height z of the vector potential increases. As the sampling height increases, then so only the dominant corners remain. However, the majority of the significant corners persist over the full set of scales.

To provide some comparison, we have provided some experimentation with the SUSAN corner detector[22]. Figure 3(a) is the original INRIA office image. In Figure 3(b) we show the result of applying the algorithm reported in this paper, while Figure 3(c) shows the result of applying the SUSAN corner detector. The results obtained with our algorithm are generally cleaner, and there are fewer false positives. There are also some interesting qualitative differences in the detected corners. For instance in our algorithm, the meeting of the line-like horizontal bars and thicker vertical bars of the window are detected as single junctions. In the case of SUSAN, double corners are returned. The result of our algorithm is more perceptually intuitive and may prove more useful for higher level matching problems.



(a) Scale = 0 (b) Scale = 1 (c) Scale = 2

(d) Scale = 0 (e) Scale = 1 (f) Scale = 2

Figure 2: Corner detection in different scales

7 Performance analysis

Our final piece of experimental work is aimed at measuring the noise sensitivity of our corner detector and comparing it with that of the SUSAN corner detector. To realise this comparison, we have generated a synthetic image of a cog-wheel and have added salt-and-pepper noise with known proportion. By varying the number of spikes on the circumference of the cog, we can vary the opening angle of the corners. The synthetic figure provides ground-truth data in which the number of target corners is known. We focus on two aspects of our corner detector. The first of these is the scale-sensitivity of the corner detection process. The second is its noise-sensitivity when compared to the SUSAN corner detector.

As mentioned before, our method can detect corners at different scales. To evaluate the performance of the algorithm, we have applied the corner detector to the vector potential sampled at various heights above the image plane. Figure 4 shows the probability of correctly detected corners as a function of different scales. The different curves are for different angles. The main conclusion that can be drawn from this plot is that corner detection at small scale(such as scale 0 and scale 1) almost always outperforms that at large scale.

The second aspect of our sensitivity study is to provide some comparison with the SUSAN corner detector under conditions of controlled noise. The plots in Figure 5 show the probability of both correctly detected corners and false corners as a function of the probability of added noise. In each case, the solid curve is the result of the proposed

(a) Original image (b) Corners from new method (c) Corners from SUSAN

Figure 3: Corner detection results

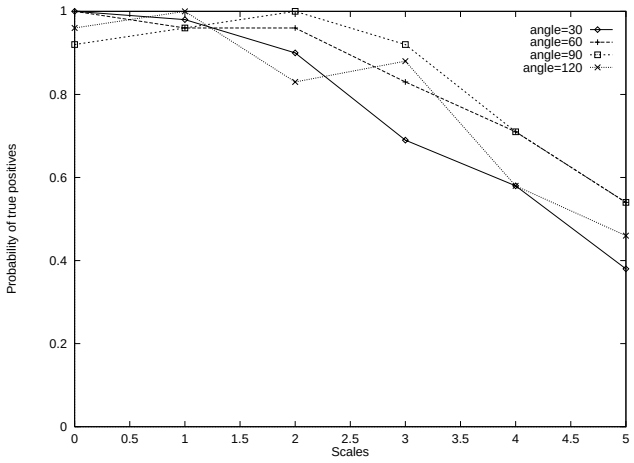


Figure 4: Comparison in different scales

algorithm, while the dotted curve is for the SUSAN corner detector. From the plots, we can draw the conclusion that the proposed algorithm consistently outperforms the SUSAN corner detector in the sense that it returns fewer false positives. The difference between true positives detected by the two methods are not so different. For small angles, such as 30 degrees, the SUSAN corner detector is better than ours. But for large angles, such as 60 and 90 degrees, the proposed method is outperforms that of the SUSAN.

8 Conclusions

In this paper we have presented a corner detection method that is based on the topographic analysis of a vector-potential image representation. According to our representation, corners are saddle-points where saddle-ridges and saddle-valleys intersect. We exploit this property to develop a simple template-based corner detector. We align the templates in the direction of the vector potential. The test for a corner is based on a local minimum

in the vector potential direction and a local maximum in the orthogonal direction. We impose a further test on consistency which is based on a significant joint change in the x and y components of the vector potential. This ensures that there is a significant change in boundary orientation.

Experimental results reveal the method to be an effective tool when compared with the SUSAN corner detector. The proposed method is less sensitive to noise compared with the SUSAN corner detector. Furthermore, our method offers the dual advantages of providing a multi-scale representation and of delivering a more perceptually meaningful junction representation for line-structures.

There are a number of ways in which the ideas presented in this paper could be developed. We clearly have a means of computing a curvature scale-space. In this respect our work is similar to that of Brady and Asada[9] and Mokhtarian and Mackworth[17]. Our next step is to emulate this work by investigating how the new corner representation can be used for shape matching and recognition.

References

[1] A.Rosenfeld and J. S. Weszka. An improved method of angle detection on digital curves. *IEEE Tran. Comput.*, 24:940–941, 1975.

[2] H. L. Beus and S. S. H. Tiu. An improved corner detection algorithm based on chain-code plane-curves. *Pattern Recognition*, 20(20):291–296, 1987.

[3] J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Matchine Intelligence*, 8(6):679–698, 1986.

[4] M. F. Costabile, C. Guerra, and G.G. Pieroni. Matching shapes: a case study in time varying images. *Computer Vision, Graphics and Image Processing*, 29:296–310, 1985.

[5] Andrew D. J. Cross and Edwin R. Hancock. Scale-space vector field for feature analysis. In *IEEE Computer Society Conferece on Computer Vision and Pattern Recognition*, pages 738–743, San Juan,Puerto Rico, 1997. IEEE Computer Society.

[6] R. Deriche and G. Giraudon. A computational approach for corner and vertex detection. *Int. J. Comput. Vision*, 10:101–124, 1993.

[7] L. Dreschler and H. Nigel. Volumetric model and 3D trajectory of a moving car derived from monocular TV-frame sequence of a street scene. *Proceedings of IJCAI*, pages 692–697, 1992.

[8] H. Freeman and L. S. Davis. A corner finding algorithm for chain code curves. *IEEE Tran. Compt.*, 26:297–303, 1977.

[9] Asada H. and Brady M. The curvature primal sketch. *IEEE Trans. on Pattern Analysis and Matchine Intelligence*, 8(1):2–14, 1986.

[10] M. H. Han and D. Jang. The use of maximum curvature points for the recognition of partially occluded objects. *Pattern Recognition*, 23:21–33, 1990.

[11] R.M. Haralick, L.T. Watson, and T.J. Laffey. The topographic primal sketch. *IJRR*, 2:50–72, 1983.

[12] L. Kitchen and A. Rosenfeld. Gray-level corner detection. *Pattern Recognition Letters*, 1:95–102, 1982.

[13] J. Koplowitz and S. Plante. Corner detection for chain coded curves. *Pattern Recognition*, 28(6):843–852, 1995.

[14] H. C. Liu and M. D. Srinath. Corner detection from chain-code. *Pattern Recognition*, 23(1-2):51–68, 1990.

[15] H. C. Liu and M. D. Srinath. Partial classification using contour matching in distance transformation. *IEEE Transactions on Pattern Analysis and Matchine Intelligence*, 12:1072–1079, 1990.

[16] R. Mehrotr, S. Nichani, and N. Ranganathan. Corner detection. *Pattern Recognition*, 23(11):1223–1233, 1990.

[17] F. Mokhtarian and A. K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Matchine Intelligence*, 14(8):789–805, 1992.

[18] J.A. Noble. Finding corners. *IVC*, 6(2):121–128, 1988.

[19] K. Rangarajan, M. Shah, and D. V. Brackle. Optimal corner detection. *Computer Vision, Graphics and Image Processing*, 48:230–245, 1989.

[20] A. Rosenfeld and E. Johmston. Angle detection on digital curves. *IEEE Tran. Comput.*, 22:875–878, 1973.

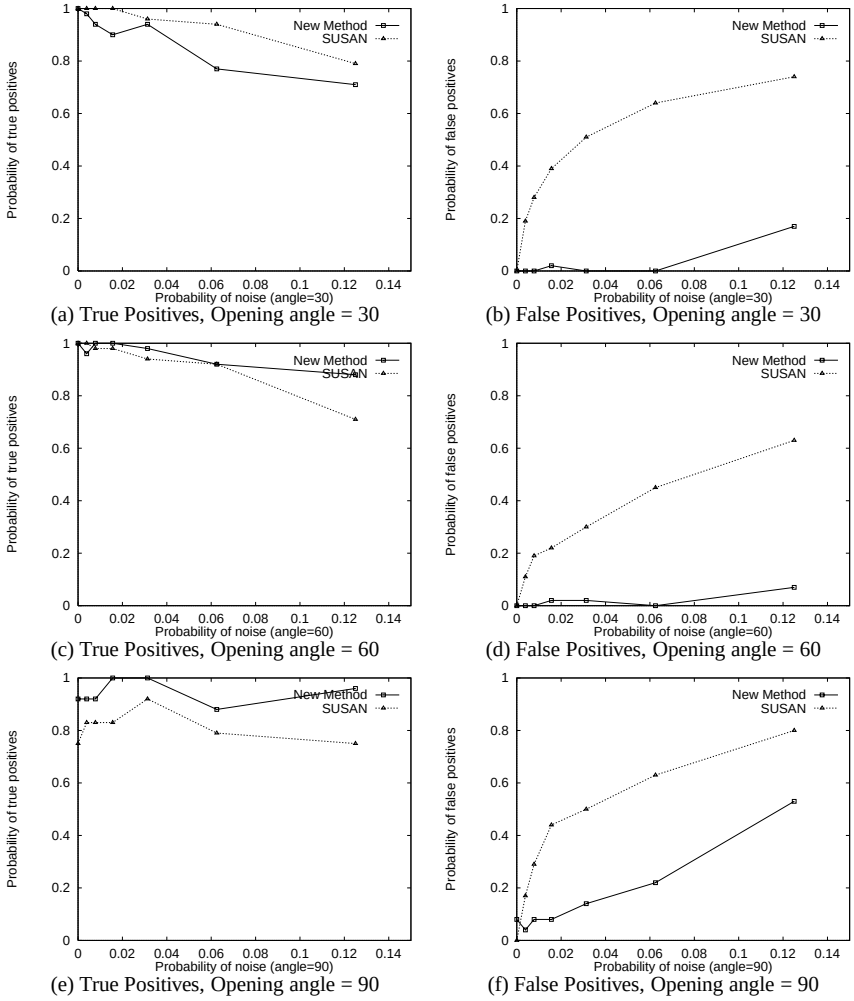
[21] A. Singh. Gray level corner detection - A generalization and a robust real-time implementation. *Computer Vision, Graphics and Image Processing*, 51(1):54–69, 1990.

[22] S.M. Smith and J.M. Brady. SUSAN - a new approach to low level image processing. *Int. Journal of Computer Vision*, 1997. In publication.

[23] D. M. Tsai. Boundary-based corner detection using neural networks. *Pattern Recognition*, 30:85–97, 1997.

[24] H. Wang and M. Brady. Real-time corner detection algorithm for motion estimation. *IVC*, 13(9):695–703, 1995.

[25] X. Xie, R.Sudhakar, and H. Zhuang. Corner detection by a cost minimization approach. *Pattern Recognition*, 26(8):1235–1243, 1993.



(a) True Positives, Opening angle = 30
(b) False Positives, Opening angle = 30
(c) True Positives, Opening angle = 60
(d) False Positives, Opening angle = 60
(e) True Positives, Opening angle = 90
(f) False Positives, Opening angle = 90

Figure 5: Noise sensitivity for various opening angles(Solid curve, our method; dotted curve, SUSAN)