

DDC FILE COPY AD.A059124

GZ-RMN-GZF

USCIPI Report 830

LEVEL



12

UNIVERSITY OF SOUTHERN CALIFORNIA

QUANTITATIVE METHODS OF EDGE DETECTION

Escandar by

Ikram Escandar Abdou

July 1978

Image Processing Institute
University of Southern California
Los Angeles, California 90007

Sponsored by

Advanced Research Projects Agency
Contract No. F-33615-76-C-1203
ARPA Order No. 3119

This document has been approved for public release and sale; its distribution is unlimited.

DDC
SEP 24 1978
INSISTIVE
File



IMAGE PROCESSING INSTITUTE

78 09 19 024

QUANTITATIVE METHODS OF EDGE DETECTION

by

Ikram Escandar Abdou

July 1978

Image Processing Institute
University of Southern California
Los Angeles, California 90007

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by the Wright Patterson Air Force Base under Contract F-33615-76-C-1203, ARPA Order No. 3119.

The views and conclusions in this document are those of the author and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U.S. Government.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Image Processing Institute
University of Southern California
Los Angeles, California 90007

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP

3. REPORT TITLE

8 QUANTITATIVE METHODS OF EDGE DETECTION,

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

9 Technical Report, July 1978

(First name, middle initial, last name)

10 Ikram Escandar/Abdou

6. REPORT DATE

11 July 1978

7a. TOTAL NO. OF PAGES

176

7b. NO. OF REFS

32

8a. CONTRACT OR FILE NO.

15 F33615-76-C-1203

8b. PROJECT NO.

ARPA Order 3-3119

9. ORIGINATOR'S REPORT NUMBER(S)

14 USCIPI Report-830

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT

Approved for release: distribution unlimited

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, Virginia 22209

13. ABSTRACT

Most local operators used in edge detection can be modelled by one of two methods: edge enhancement/thresholding and edge fitting. This dissertation presents a quantitative design and performance evaluation of these methods. The design techniques are based on statistical detection theory and deterministic pattern recognition classification procedure. The performance evaluation methods developed include: (a) deterministic measurement of the edge gradient amplitude; (b) comparison of the probabilities of correct and false edge detection; and (c) figure of merit computation. The design techniques developed are used to optimally design a variety of small and large mask edge enhancement/thresholding operators. A performance comparison is given between these edge detectors. A new edge fitting algorithm is introduced. The new algorithm is derived in the discrete domain, this allows a direct optimization of the operator's performance. The advantages of the new algorithm are better performance with real world pictures and less sensitivity to signal-to-noise ratio.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Key Words: Edge Detection, Statistical Detection Theory, Pattern Classification, Figure of Merit, Edge Fitting, Orthogonal Transformation						

UNCLASSIFIED

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to Professor W. K. Pratt for his valuable guidance and enthusiastic encouragement throughout the various stages of research that led to accomplishing this dissertation. Sincere thanks are also extended to Professor R. Nevatia and Professor E. K. Blum, the other committee members, for their assistance and contributions. Special appreciation is due to Professor H. C. Andrews and the staff of the Image Processing Institute for their support given in preparation of this work. Finally, the author wishes to thank Amy Yiu for her superb job in typing and bringing this work to completion.

ACCESSION for	
NTIS	White Section <input checked="" type="checkbox"/>
DOC	Buff Section <input type="checkbox"/>
UNCLASSIFIED	<input type="checkbox"/>
CLASSIFIED	<input type="checkbox"/>
BY	
DISPATCH/INTEGRITY CODES	
SERIAL	
A	

TABLE OF CONTENTS

	Page
ACKNOWLEDGEMENTS	ii
LIST OF FIGURES	vii
LIST OF TABLES	x
Chapter	
1. INTRODUCTION	1
1.1 Edge Detection Techniques	1
1.2 Edge Detection Evaluation	7
1.3 Organization of Dissertation	8
2. REVIEW OF EDGE DETECTION OPERATORS	10
2.1 Edge Enhancement/Thresholding Methods	12
2.1.1 Simple Differential Operators	14
2.1.2 Template Matching Operators	17
2.2 Edge Detectors Performance, Case of Ideal Edge	21
2.2.1 Case of Central Edge with Orientation ϕ	21
2.2.2 Case of a Fixed-Orientation Edge with Varying Displacement	24
2.3 Edge Fitting Method-Hueckel's Algorithm	31
2.4 Conclusion	35
3. STATISTICAL MODEL FOR EDGE DETECTION	36
3.1 Edge Detection as a Hypothesis-Testing Problem	36

TABLE OF CONTENTS (CONT'D)

Chapter		Page
	3.2 Edge Detection Performance, Case of Ideal Edge Plus Noise	41
	3.2.1 Simple Differential Operators	43
	3.2.2 Template Matching Operators	49
	3.3 Estimation of the Edge Orientation	51
	3.4 Conclusion	55
4.	EDGE DETECTION AS A PATTERN CLASSIFICATION PROBLEM	57
	4.1 Training Methods for Pattern Classifiers	58
	4.2 The Ho-Kashyap Algorithm	61
	4.3 Application of the Ho-Kashyap Algorithm to Edge Detection	64
	4.4 Conclusion	65
5.	FIGURE OF MERIT COMPARISON OF EDGE DETECTORS	68
	5.1 Figure of Merit Concepts	69
	5.2 Experimental Results	74
	5.3 Conclusion	79
6.	NEW EDGE ENHANCEMENT/THRESHOLDING METHODS	82
	6.1 Effect of Changing Mask Size	83
	6.2 Use of Weighted Masks	93
	6.3 Use of Adaptive Thresholding	97
	6.4 Conclusion	100

TABLE OF CONTENTS (CONT'D)

Chapter	Page
7. A NEW EDGE FITTING ALGORITHM	102
7.1 One-Dimensional Edge Fitting	103
7.2 Two-Dimensional Edge Fitting	108
7.3 Performance Evaluation	110
7.4 Conclusion	117
8. CONSLUSION AND FURTHER WORK	122
 APPENDIXES	
A Analysis of the Hueckel Algorithm	124
A.1 A Review of the Hueckel Algorithm	124
A.2 Effect of Truncation of the Orthogonal Expansion	127
A.3 Effect of Inaccuracy in the Minimization Procedure	128
B Orthogonal Transformation in Edge Detection	135
B.1 Edge Model in the Discrete Fourier Domain	136
B.2 Line Models in the Discrete Fourier Domain	139
B.3 Performance Analysis of the Discrete Fourier Transform Edge Detector	146
C Derivations of Eqs. 3.29, 3.31 and 3.32	149
D The Herskovits Algorithm	151

TABLE OF CONTENTS (CONT'D)

APPENDIXES	Page
E Experimental Results	154
REFERENCES	163

LIST OF FIGURES

Figure		Page
2.1	Edge model	11
2.2	Edge enhancement/thresholding edge detection system	13
2.3	Image subregions	15
2.4	Template matching operators	18
2.5	Elements of the l 'th template	20
2.6	Edge models for edge orientation intensity analysis	22
2.7	Edge gradient amplitude response as a function of actual edge orientation for 2x2 and 3x3 operators	25
2.8	The detected edge orientation as a function of actual edge orientation for 2x2 and 3x3 operators	26
2.9	Edge models for edge displacement sensitivity analysis	27
2.10	Edge gradient amplitude response as a function of edge displacement for 2x2 and 3x3 operators	29
2.11	Hueckel's edge-line model	33
3.1	Typical conditional probability density functions of edge enhancement	39
3.2	Probability of detection versus probability of false detection for simple differential operators	47
3.3	Probability of detection versus probability of false detection for template matching operators	52

LIST OF FIGURES (CONT'D)

Figure	Page
5.1 Types of edge detection errors	70
5.2 Figure of merit test image geometry for diagonal edge	73
5.3 Figure of merit as a function of signal-to-noise ratio for simple differential operators	75
5.4 Figure of merit as a function of signal-to-noise ratio for template matching operators	77
5.5 Edge maps for 2x2 and 3x3 operators	80
6.1 Extended masks for the Prewitt and the 3-level operators	84
6.2 Probability of detection versus probability of false detection for extended Prewitt and 3-level operators	86
6.3 Edge gradient amplitude response as a function of edge displacement for extended Prewitt and 3-level operators	88
6.4 Figure of merit as a function of signal-to-noise ratio for extended Prewitt and 3-level operators	89
6.5 Edge maps for extended Prewitt operator, vertical test image with SNR=1	92
6.6 Pyramid operator	95
6.7 Edge gradient amplitude response as a function of edge displacement for weighted 7x7 operators	96
6.8 Figure of merit as a function of signal-to-noise ratio for weighted 7x7 operators	99
7.1 One-dimensional edge model	104
7.2 Two-dimensional edge model	111

LIST OF FIGURES (CONT'D)

Figure		Page
7.3	Edge fitting normalized error $\sqrt{E_{\min}}/\Delta$, as a function of actual edge orientation	113
7.4	Edge fitting normalized error $\sqrt{E_{\min}}/\Delta$, as a function of edge displacement	114
7.5	Edge fitting normalized error $\sqrt{E_{\min}}/\Delta$, as a function of actual edge orientation for noisy edges	116
7.6	Figure of merit as a function of signal-to-noise ratio for edge fitting operator	118
7.7	Edge maps for the edge fitting operator, diagonal test image with SNR=1	120
A.1	Ideal and reconstructed edge and line models	129
A.2	Figure of merit as a function of signal-to-noise ratio for the Hueckel operator	133
B.1	Edge models for the discrete Fourier transform	137
B.2	Fourier coefficients for a 5x5 edge	140
B.3	A one-pixel-line model	141
B.4	Fourier coefficients for a 5x5 line	144
B.5	Detection of a rotated vertical line	145
C.1	The equation $A = X + Y $	150
E.1	Examples of edge maps, girl picture	155
E.2	Examples of edge maps, airport picture	157
E.3	Examples of edge maps, tank picture	159

LIST OF TABLES

Table		Page
3.1	Mean Vector and Covariance Matrix of Differential Gradient Operators	44
3.2	Mean Vector and Covariance Matrix of Template Matching Operators for a vertical Edge	50
4.1	Threshold Level and Error Probabilities for Ho-Kashyap Design Procedure	66
6.1	The Ratio $(\frac{a}{\sigma_r})$ for Weighted Masks	98

Chapter 1

Introduction

Image edges can be defined as local changes or discontinuities in an image attribute such as luminance, tristimulus value, or texture [1]. These changes are important in the analysis of images because they often provide an indication of the physical extent of objects within the image. An operator used to detect these changes is called an edge detector. This operator transforms an image into a binary array containing ones where the magnitude of the discontinuity is significant and zeros elsewhere. The binary array obtained is usually called an edge map. This transformation is useful in image understanding systems, because while the edge map retains much of the basic structure of the image, less computational effort is required for analysis as compared to the original image.

1.1 Edge Detection Techniques

There are many techniques which can be used in edge detection. These include simple differential operators, template matching, least square edge fitting, and

techniques based on statistical detection theory. There are also many heuristic methods developed for edge detection. A complete survey of all edge detectors is not a simple task, and can even be confusing. Hence, only a group of the most useful operators will be discussed in the following sections.

Linear differential operators are commonly employed in edge detection. In this method, edges are enhanced by convolving the image with a set of discrete differential operator masks. A corresponding edge map is obtained by thresholding some function of the outputs of these masks. One of the differential operators used is the gradient. The gradient is approximately calculated by convolving the image with two masks that measure the pixels luminance change in any two orthogonal directions. The sum of the squares of the masks output is a measure of the gradient magnitude squared. Roberts has used 2×2 masks to compute the luminance difference across the diagonals [2], while Prewitt [3] and Sobel [4] have used 3×3 masks to measure the difference in the horizontal and vertical directions. Another differential operator, which has been used in edge enhancement, is the Laplacian operator. Examples of the Laplacian masks are given in [1, 3]. However, since the Laplacian operator is more sensitive to points and lines than to edges [5], it is not an efficient method for edge detection. In general, all of the linear differential

operators have the advantage of using simple mathematical formulas which require short computation time. Their major disadvantage is their sensitivity to noise. One method to improve the performance of differential operators, in the presence of noise, is to increase the masks size. This can be noticed in comparing the performances of the Roberts and the Sobel operators. Another, and rather better method, is to design edge detectors taking into consideration the effect of noise. This leads to using template matching in edge detection.

The problem of edge detection can be reformulated as follows [1]: given a subregion of the image, find one member of a finite group of templates representing edges and no edges, such that this member matches the subregion as close as possible and label the subregion accordingly. Matching is usually measured in terms of the mean square difference between the subregion and the templates. Calculation can be simplified by expanding the mean square difference and neglecting the slowly varying terms. The remaining term is the cross correlation between the subregion and the templates. This term should be maximum for the best match. Cross-correlation template matching has been used in edge detection. One of the template matching operators was introduced by Prewitt [3]. The Prewitt method aimed at finding a better evaluation of the gradient operator by using a set of oriented edges and

searching sequentially at each point for the best match. In this method, gradient magnitude is equated with the maximum response, and direction is taken parallel to the orientation of the corresponding detector [3]. The templates correspond to horizontal, vertical and diagonal edges. Other forms of templates were later introduced by Kirsch [6] and Robinson [7]. The basic advantages of these operators are that they can be implemented with a relatively small computation effort. In addition, proper choice of the template coefficients gives almost optimum performance. However, optimum performance can never be achieved since the number of templates used is always finite. A different approach to achieve optimum performance was later introduced by Hueckel.

In Hueckel's algorithm [8], edges are detected by fitting circular subregions of the image to ideal edge models. If the fit is sufficiently accurate, an edge is assumed to exist with the same parameters as the ideal edge model. The edge model used is a two-dimensional step in a circular disc. The parameters of this model are the luminance levels, the edge orientation and distance from the center. The accuracy of edge fitting is measured in terms of the mean square error criterion. Hueckel introduced a polar Fourier expansion and used the first eight coefficients in the minimization procedure. Although this approximation simplifies the computation needed, it

affects the accuracy of the minimization procedure. Hueckel has not provided any evaluation of this problem.

Another method to achieve optimum edge detection is to introduce statistical detection theory concepts. In the statistical model, images are considered to be the sum of two components; the first is an ideal image in which edges of different orientations and heights are distributed, while the second consists of a random additive noise. For this model, edge detectors are designed to achieve an optimum probability of correct decisions. Griffith has used this approach in the analysis of scenes consisting of prismatic solids. He introduced a detailed study of the distortion and noise affecting the image, and implemented a decision procedure based on computing the probability that a line representing a real edge is centered in and traverses some long narrow band. But, the computation of this probability was a difficult task, and the final results were based on many unjustified approximations [9]. A different approach to statistical edge detection was proposed by Yakimovsky [10]. In this approach, two adjacent regions of the image are tested; first assuming that they have the same average luminance, and then assuming that they have two different luminance levels. Maximum likelihood estimates in both cases are compared, and an edge is indicated if it is more likely that the regions have two different luminance levels. A

disadvantage of the Griffith and Yakimovsky algorithms is that they are designed to detect edges of a certain orientation. They are less sensitive to edges with other orientations. To avoid this problem, the operator is usually applied with enough orientations to give uniform response. The different results are then combined to form the edge map.

A completely different approach to edge detection is to use the a priori knowledge of the image objects in searching for their boundaries. Examples can be found in the work of Kelly [11] and Chow [12]. Kelly introduced a program for extracting an accurate outline of a man's head from a digital picture [11]. His method consisted of three steps. First, a new digital picture was prepared from the original; the new picture is smaller and has less detail. Then edges of objects are located in the reduced picture. Finally, the edges found in the reduced picture are used as a plan for finding edges in the original picture. Chow studied the problem of detecting the boundary of the human heart in a cineangiogram [12]. He assumed that the probability distribution of any small region of the picture that contains only object or only background is unimodal, and a region that contains both object and background will be a mixture of the two distributions. The unimodal distributions are assumed to be Gaussian. Starting from these assumptions, Chow's algorithm examines the

probability distribution of the image subregions. If the standard deviation is large, the probability distribution is fitted to a bimodal Gaussian. The bimodality is measured by computing the valley-to-peak ratio. If this ratio is high, the points in the subregion are classified as a part of the object or the background depending on their intensity. Although the Chow algorithm is successful in determining the boundary in single-object scenes, it is not directly extendable to scenes with many objects. This later case is more important in scene analysis. Because the previous operators are limited in their applications, they will not be considered further in this dissertation.

1.2 Edge Detector Evaluation

Another field of study in edge detection, which has not been given enough consideration, is the performance evaluation of edge detectors. As stated in reference [1], this evaluation is difficult because of the large number of proposed methods, the difficulties in determining the best parameters associated with each technique, and the lack of definite performance criteria. One method for edge detection evaluation was suggested by Fram and Deutsch [13]. In this method, a test image in the form of ideal ramped edge with additive Gaussian noise is used to evaluate the performance of edge detectors suggested by Hueckel, Macleod, and Rosenfeld. Two parameters are used

in this evaluation, the first is the maximum likelihood estimate of the ratio between the number of correct detections of edges and the total number of detected edges. The practical significance of the second parameter is not clear. The results are compared with human ability to perceive edges. In this experiment, the results obtained with the Hueckel operator appear to be inferior. This can be partially explained by the fact that the Hueckel internal parameters used are far from the optimum choice. Another method for measuring the performance of edge detectors was given by Pratt [1]. This method uses a figure of merit which is sensitive to the different kinds of errors encountered in edge detection: missing or displacing a true edge and the false detection of noise. The figure of merit introduced has been used to measure the optimum performance of the Roberts, Sobel, Kirsch, and compass gradient operators in the case of an artificial image of a vertical edge with additive Gaussian white noise. The experiment shows that the Kirsch and the Sobel operators have relatively high figures of merit followed by the compass gradient operator and finally the Roberts operator. These results agree with the visual data.

1.3 Organization of Dissertation

In the previous survey it should be noticed that while there are many operators that can be used in edge

detection, the effort given to the comparison and evaluation of these operators has not been sufficient. A quantitative evaluation of the edge detectors is needed if these operators are to be efficiently used as a part of an image understanding system. The following chapters will be devoted to the introduction of quantitative methods into edge detection problems. In Chapter 2, a detailed discussion of the basic edge detection operators, used in this dissertation, is given. An image model is developed in Chapter 3, and used to evaluate the performance of these edge detection operators. In Chapter 4, edge detection is formulated as a pattern classification problem, and a least square error algorithm is used to determine the edge detectors parameters. The figure of merit derived by Pratt is used in Chapter 5 to evaluate the performance of the different operators in the case of vertical or diagonal edges. The results obtained in these chapters are used in the improvement of existing operators and in the introduction of new methods for edge detection. These are given in Chapters 6 and 7, respectively. In Chapter 8, some final conclusions are presented.

Chapter 2

Review of Edge Detection Operators

The edge detectors of interest in this dissertation can be defined as local operators which are able to detect image discontinuities without any a priori knowledge of the image content. These local operators are useful as a first step in many image understanding systems. Most of the local edge detectors can be classified into two basic groups. The first is the edge enhancement/thresholding methods that includes the use of simple differential operators and template matching. The second is the edge fitting technique. For purposes of design and analysis, the input to the edge detector is assumed to be an ideal ramp edge as shown in Figure 2.1. The function represented in this figure is usually the luminance attribute. Parameters that describe this edge are its location, orientation, edge width and height. These parameters are to be estimated by the edge detector. One of the factors which determine the edge detector's performance, is the operator's accuracy in estimating the edge parameters.

In this chapter, a detailed analysis of some of the edge detection operators is given. Section 2.1 reviews the

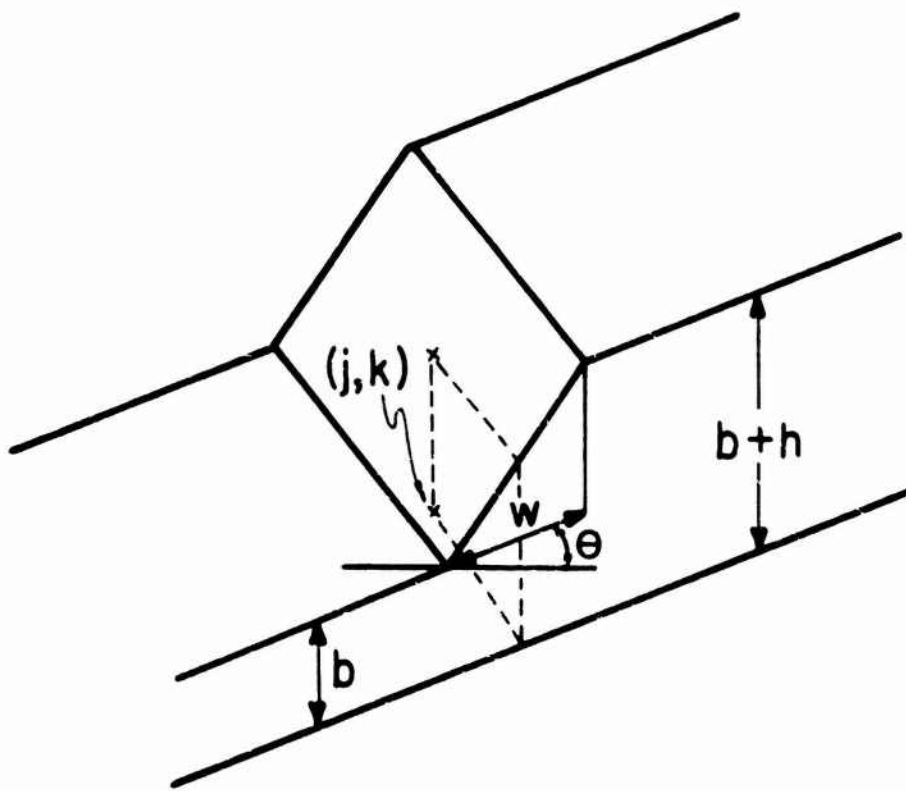


Figure 2.1. Edge model

edge enhancement/thresholding operators. Section 2.2 evaluates the edge detectors performance using an ideal edge model. Section 2.3 discusses the edge fitting technique.

2.1 Edge Enhancement/Thresholding Methods

The edge enhancement/thresholding technique can be represented by the block diagram shown in Figure 2.2. In this model, the image $F(j,k)$ is first convolved with a set of linear spatial operators $\{H_i(j,k)\}$, the output $G_i(j,k)$ is given by

$$G_i(j,k) = H_i(j,k) \bullet F(j,k) \quad (2.1)$$

where $i = 1, 2, \dots, m$. A nonlinear function of the set $\{G_i(j,k)\}$ is then calculated. The output $A(j,k)$ is described by the equation

$$A(j,k) = g(G_1(j,k), G_2(j,k), \dots, G_m(j,k)) \quad (2.2)$$

Typical forms of the function $g(\cdot)$ are the sum of squares, the square root, the magnitude, the maximum or combinations of these functions. The output $A(j,k)$ is a measure of the discontinuity at the center of the convolving masks; it can be used to form a grey-level edge map. In order to improve edge visibility, and to reduce the edge map complexity at the same time, the grey-level edge map is compared with a threshold t , and an edge is detected if

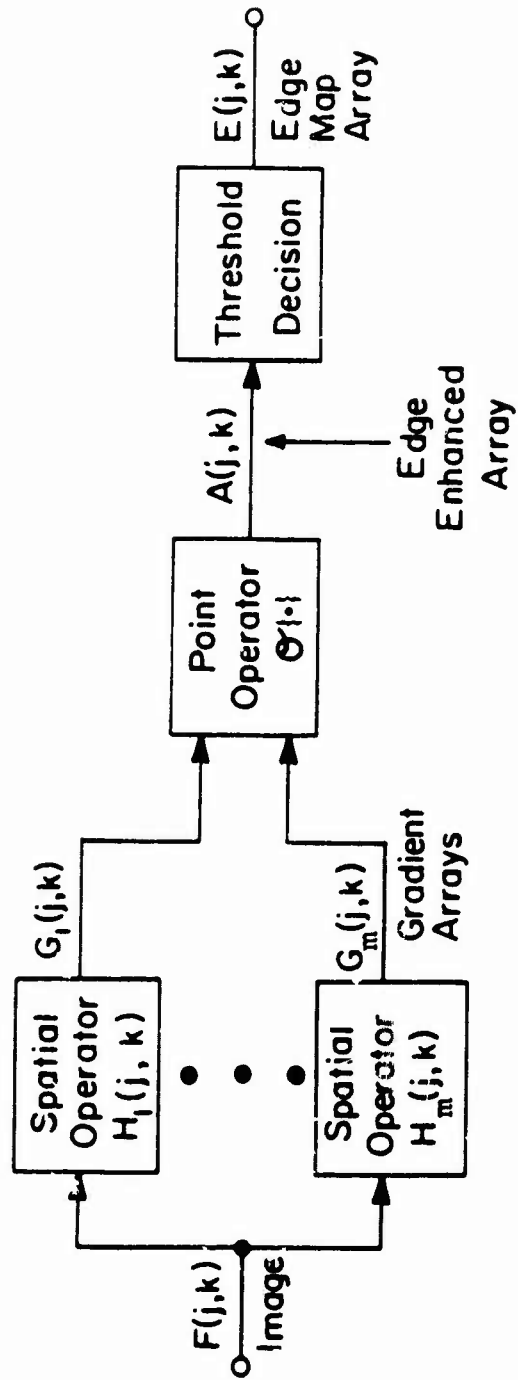


Figure 2.2. Edge enhancement/thresholding edge detection system

$$A(j,k) \geq t \quad (2.3a)$$

while if

$$A(j,k) < t \quad (2.3b)$$

the decision is no edge. The threshold t defines the resulting edge map; if it is chosen too high, then low-amplitude changes will not be detected, and if it is chosen too low, noise can be falsely detected as edges [1].

If an edge is detected, it is often useful to determine its orientation and height. This information can be obtained from the set $\{G_i(j,k)\}$, as will be shown later.

After this general introduction to the edge enhancement/thresholding technique, some important examples of the simple differential operators and template matching operators will be given.

2.1.1 Simple Differential Operators

This group of edge detectors includes the Roberts [2], the Sobel [4], and an operator suggested by Prewitt [3]. The Roberts operator is applied on 2×2 subregions of the image as sketched in Figure 2.3a. The output $A(j,k)$ is given by

$$A(j,k) = \left[(f_2 - f_3)^2 + (f_1 - f_4)^2 \right]^{\frac{1}{2}} \quad (2.4)$$

f_1 f_2

 f_3 f_4

a. 2x2 Subregion

f_1 f_2 f_3

 f_4 f_5 f_6

 f_7 f_8 f_9

b. 3x3 Subregion

Figure 2.3. Image subregions

Equation 2.4 can be viewed as two convolutions

$$X(j,k) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \odot F(j,k) \quad (2.5a)$$

$$Y(j,k) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \odot F(j,k) \quad (2.5b)$$

followed by the nonlinearity

$$A(j,k) = \left[(X(j,k))^2 + (Y(j,k))^2 \right]^{1/2} \quad (2.6)$$

Roberts has also introduced a magnitude operator, in which the discrete gradient is alternatively calculated as

$$A(j,k) = |X(j,k)| + |Y(j,k)| \quad (2.7)$$

In both operators, an edge is detected if $A(j,k) \geq t$, where t is a given threshold. If an edge is detected, its orientation is given by

$$\theta(j,k) = \frac{\pi}{4} + \tan^{-1} \left(\frac{Y(j,k)}{X(j,k)} \right) \quad (2.8)$$

The angle $\theta(j,k)$ is measured with respect to the horizontal axis.

Approximations of the discrete gradient function by 3×3 operators were given by Prewitt [3] and later by Sobel [4]. These operators are applied on 3×3 subregions of the image as sketched in Figure 2.3b. The outputs $X(j,k)$ and

$Y(j,k)$ are given by

$$X(j,k) = \begin{bmatrix} 1 & 0 & -1 \\ c & 0 & -c \\ 1 & 0 & -1 \end{bmatrix} \otimes F(j,k) \quad (2.9a)$$

$$Y(j,k) = \begin{bmatrix} -1 & -c & -1 \\ 0 & 0 & 0 \\ 1 & c & 1 \end{bmatrix} \otimes F(j,k) \quad (2.9b)$$

where the constants c is 1 in the Prewitt and 2 in the Sobel operator. The output $A(j,k)$ is still given by Eq. 2.6, while the edge orientation with respect to the horizontal axis is calculated by

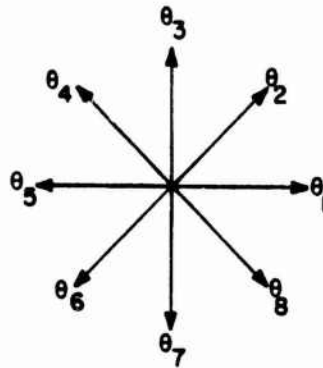
$$\theta(j,k) = \tan^{-1} \left(\frac{Y(j,k)}{X(j,k)} \right) \quad (2.10)$$

2.1.2 Template Matching Operators

The compass gradient [3], Kirsch [6], 3-level and 5-level operators [7] are examples of template matching operators. In this technique, the input image is convolved with the set of linear masks $\{H_i(j,k)\}$ shown in Figure 2.4. The outputs $\{G_i(j,k)\}$ measure the gradient components along the basic orientations. The enhanced edge is formed as the maximum of the gradient arrays. Thus

$$A(j,k) = \max \left\{ |G_1(j,k)|, |G_2(j,k)|, \dots, |G_m(j,k)| \right\} \quad (2.11)$$

If $A(j,k)$ is greater than the threshold t , an edge is



a) compass directions

		-1	3	3	-5
	-2	-1	3	0	-5
		-1	3	3	-5

i) Compass gradient

ii) Kirsch

	0	-1		0	-1
	0	-1	2	0	-2
	0	-1		0	-1

iii) 3-level

iv) 5-level

b) mask H_1

	-1	-1	3	-5	-5
	-2	-1	3	0	-5
			3	3	3

i) Compass gradient

ii) Kirsch

0	-1	-1	0	-1	-2
	0	-1		0	-1
		0	2		0

iii) 3-level

iv) 5-level

c) mask H_2

Figure 2.4. Template matching operators

detected with orientation $\theta(j,k)$ given by the compass direction of the largest gradient component. Because of the symmetry of the 3-level and 5-level masks, they can be implemented using the first four masks only.

In Chapter 1, it was mentioned that the previous four operators can be considered as cross-correlation template matching operators. This can be shown as follows; assume that it is required to match a subregion of the image with one of m templates, where the elements of the l 'th template are shown in Figure 2.5. The l 'th cross correlation is given by

$$R_l = \sum_j f_j (b + \alpha_{j,l} h) \quad (2.12)$$

The first term of Eq. 2.12 is constant for a given subregion. In addition h is proportional to $\sum \alpha_{j,l} f_j$. Thus maximizing Eq. 2.12 is equivalent to maximizing $\sum_j \alpha_{j,l} f_j$.

In this section a survey of the edge enhancement/thresholding operators has been given. It should be noticed that, because of the diversity of the operators used, it is useful to compare the performance of these operators quantitatively. There are different approaches that can be used in this comparison. One example is to compare the edge detectors outputs for a set of ideal edges. This technique will be considered in the

$$\begin{array}{ccc} b+\alpha_{1,\ell}^h & b+\alpha_{2,\ell}^h & b+\alpha_{3,\ell}^h \\ b+\alpha_{4,\ell}^h & b+\alpha_{5,\ell}^h & b+\alpha_{6,\ell}^h \\ b+\alpha_{7,\ell}^h & b+\alpha_{8,\ell}^h & b+\alpha_{9,\ell}^h \end{array}$$

Figure 2.5. Elements of the ℓ 'th template

following section. Other methods that implement statistical detection theory will be discussed in Chapter 3.

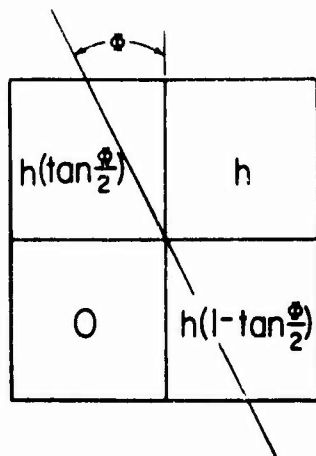
2.2 Edge Detectors Performance, Case of Ideal Edge

In this analysis, the edge model shown in Figure 2.1 is used. Here the edge is assumed to be of zero width (ideal step function). When an edge detector is applied on this edge model, the output will be determined by the edge position and orientation. To simplify the analysis, the effect of each parameter is considered separately. First, the edge is assumed to pass through the center of the edge detector with general edge orientation ϕ . Second, the edge is assumed to have a fixed orientation while its distance from the edge-detector center is varied. In both cases the outputs of the different edge detectors are evaluated.

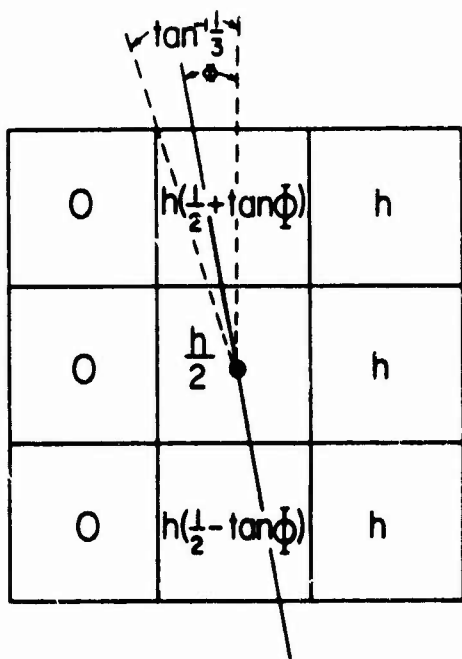
2.2.1 Case of Central Edge with orientation ϕ .

The average intensities of the different pixels, of a 2×2 and a 3×3 image subregion containing a central edge, are shown in Figure 2.6. These intensities are given as a function of the edge orientation ϕ . Because of the symmetry of the edge detectors, it is sufficient to measure the operators performance for $0 \leq \phi \leq \frac{\pi}{4}$.

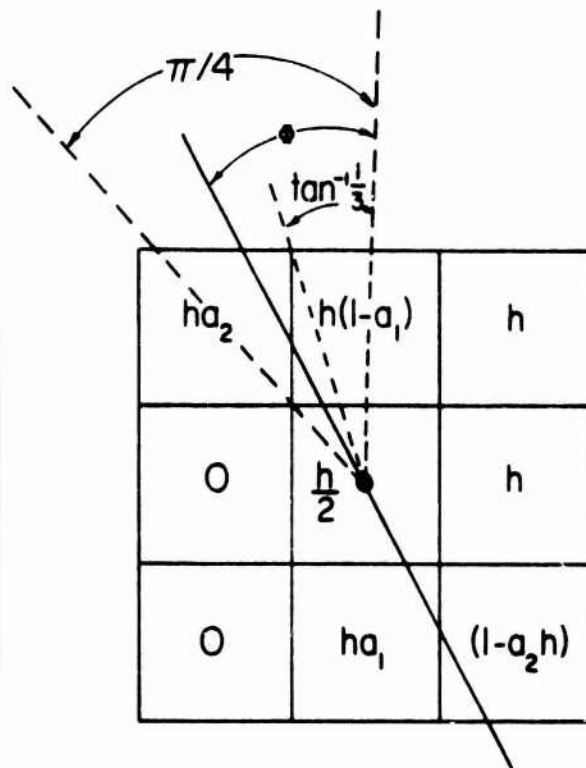
When the Sobel operator is applied on this edge model,



a) 2x2 model



b) 3x3 model
 $0 \leq \phi \leq \tan^{-1}\left(\frac{1}{3}\right)$



c) 3x3 model
 $\tan^{-1}\left(\frac{1}{3}\right) < \phi \leq \frac{\pi}{4}$
 $a_1 = \frac{(1-\tan\phi)^2}{8\tan\phi}$, $a_2 = \frac{(3\tan\phi-1)^2}{8\tan\phi}$

Figure 2.6. Edge models for edge orientation sensitivity analysis

the values of the output A^* and the estimated edge orientation are as follow.

$$A = \begin{cases} 4h[\sec(\phi)] & 0 \leq \phi \leq \tan^{-1}(\frac{1}{3}) \\ \frac{h}{4\tan(\phi)} \left[[-9\tan^2(\phi) + 22\tan(\phi) - 1]^2 \right. \\ \left. + [7\tan^2(\phi) + 6\tan(\phi) - 1]^2 \right]^{\frac{1}{2}} & \tan^{-1}(\frac{1}{3}) \leq \phi \leq \frac{\pi}{4} \end{cases} \quad (2.13)$$

$$\theta = \begin{cases} \phi & 0 \leq \phi \leq \tan^{-1}(\frac{1}{3}) \\ \tan^{-1} \left(\frac{7\tan^2(\phi) + 6\tan(\phi) - 1}{-9\tan^2(\phi) + 22\tan(\phi) - 1} \right) & \tan^{-1}(\frac{1}{3}) \leq \phi \leq \frac{\pi}{4} \end{cases} \quad (2.14)$$

Similar expressions can be obtained for the other simple differential operators.

When the Kirsch operator is applied, the values of A and θ are as follows.

$$A = \begin{cases} 12h & 0 \leq \phi \leq \tan^{-1}(\frac{1}{2}) \\ h \left[12 - \frac{(3\tan(\phi) - 1)^2}{\tan(\phi)} \right] & \tan^{-1}(\frac{1}{3}) \leq \phi \leq \tan^{-1}(\frac{1}{2}) \\ h \left[12 - \frac{(1 - \tan(\phi))^2}{\tan(\phi)} \right] & \tan^{-1}(\frac{1}{2}) \leq \phi \leq \frac{\pi}{4} \end{cases} \quad (2.15)$$

$$\theta = \begin{cases} 0 & 0 \leq \phi \leq \tan^{-1}(\frac{1}{2}) \\ \frac{\pi}{4} & \tan^{-1}(\frac{1}{2}) \leq \phi \leq \frac{\pi}{4} \end{cases} \quad (2.16)$$

Similar expressions can be obtained for the other template matching operators.

* Starting with this section, the (j,k) coordinates are dropped.

Plots of the values of A and θ for different edge enhancement/thresholding operators are given in Figures 2.7 and 2.8. In these curves, the value of A is normalized with respect to its value for a vertical edge. From these curves, it is clear that all the edge detectors are not isotropic because A varies with ϕ . This variation is smaller in the template matching operators compared to the simple differential operators. Also, the estimated edge orientation, θ is usually different from the actual orientation, ϕ . This difference is smaller for the simple differential operators than for the template matching operators. This is basically because the template matching operators measure the edge orientation in a quantized step.

2.2.2 Case of a Fixed-Orientation Edge with Varying Displacement

In this case, the edge is assumed to have a fixed orientation, while its distance to the center of the edge detector is changed. The edge orientations chosen are the vertical and the diagonal, with $\phi = 0$ and $\pi/4$, respectively. Similar results can be obtained for horizontal and $-\pi/4$ orientation edges. These are the only edge orientations for which the continuous-edge shape is preserved after sampling.

The intensities of the different pixels for a displaced vertical edge are shown in Figure 2.9. When the

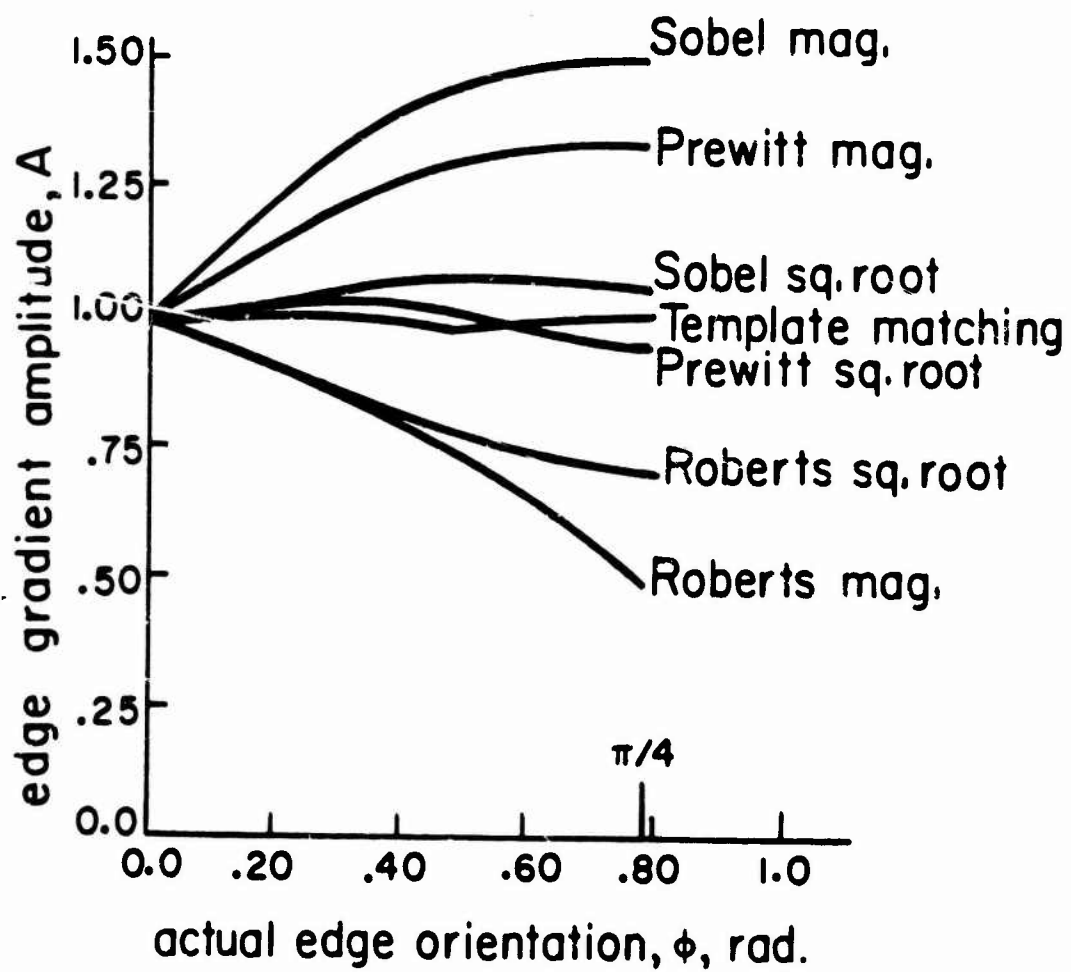


Figure 2.7. Edge gradient amplitude response as a function of actual edge orientation for 2x2 and 3x3 operators

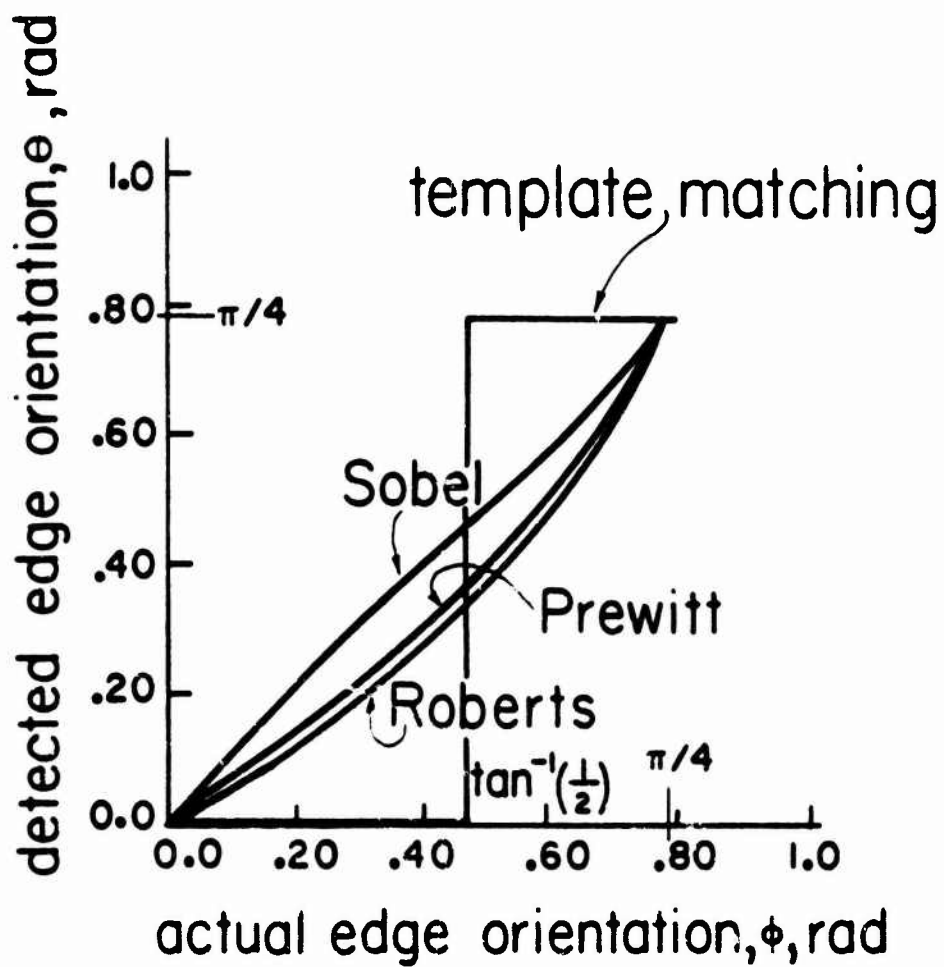
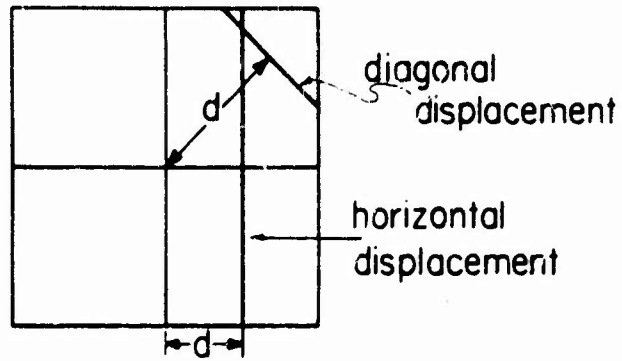
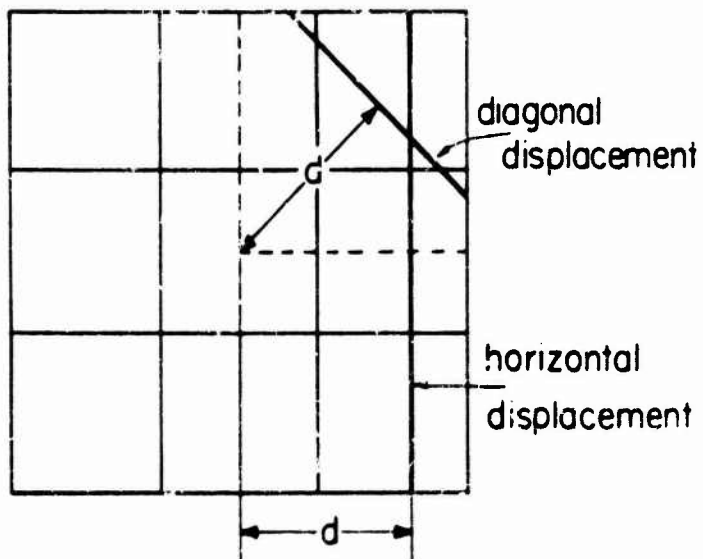


Figure 2.8. The detected edge orientation as a function of actual edge orientation for 2x2 and 3x3 operators



a) 2x2 model



b) 3x3 model

Figure 2.9. Edge models for edge displacement sensitivity analysis

Sobel operator is applied on this edge model, the value of the output A is given by

$$A = \begin{cases} 4h & 0 \leq d \leq \frac{1}{2} \\ 4h(\frac{3}{2}-d) & \frac{1}{2} \leq d \leq \frac{3}{2} \end{cases} \quad (2.17)$$

When the Kirsch operator is used, A is given by

$$A = \begin{cases} 12h(\frac{d}{2}+1) & 0 \leq d \leq \frac{1}{2} \\ 15h(\frac{3}{2}-d) & \frac{1}{2} \leq d \leq \frac{3}{2} \end{cases} \quad (2.18)$$

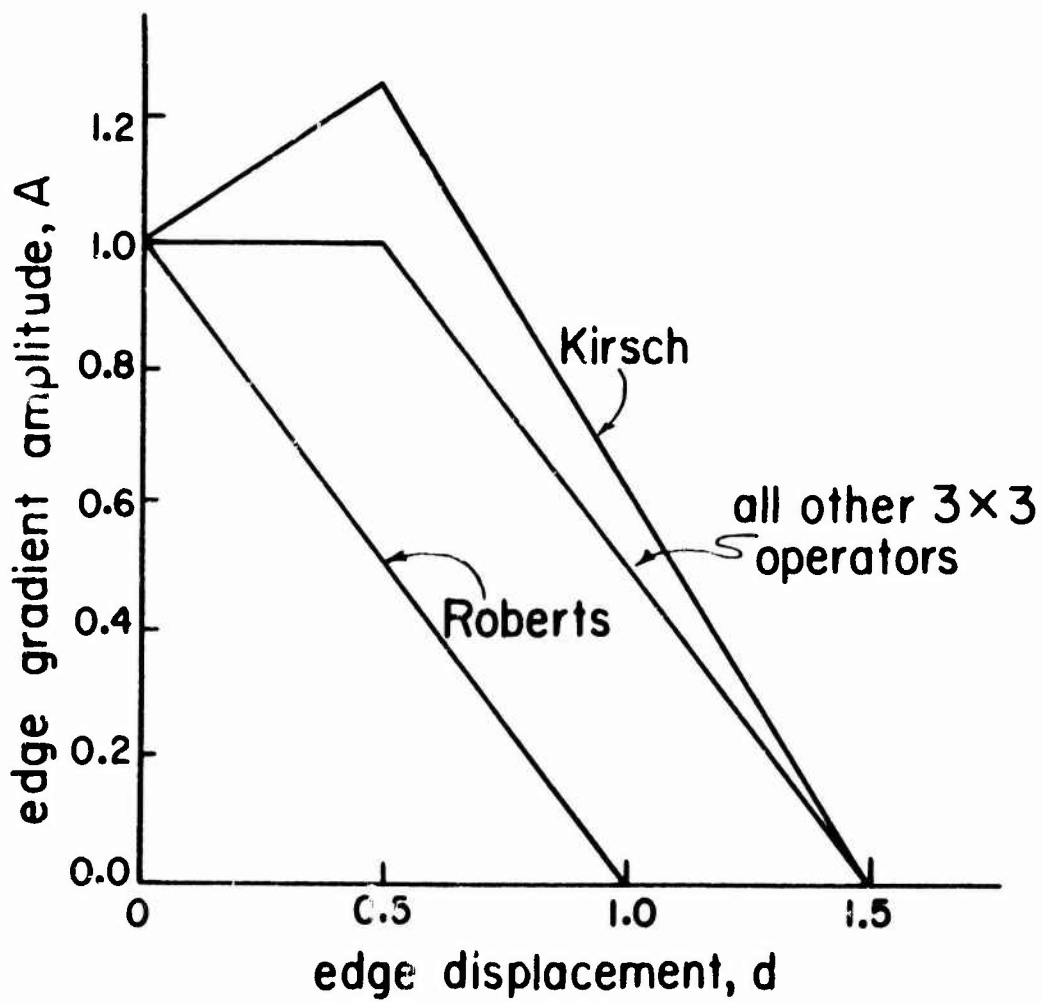
Plots of A for the different operators are shown in Figure 2.10a.

In the case of a diagonal edge, the average intensities become a second order polynomial of the distance across the diagonal. The output A for the Sobel operator is given by

$$A = \begin{cases} h(3-2d^2) & 0 \leq d \leq \frac{1}{2} \\ h[1-(d-\frac{1}{\sqrt{2}})^2+2(\sqrt{2}-d)^2] & \frac{1}{\sqrt{2}} \leq d \leq \sqrt{2} \\ h(\frac{3}{\sqrt{2}}-d)^2 & \sqrt{2} \leq d \leq \frac{3}{\sqrt{2}} \end{cases} \quad (2.19)$$

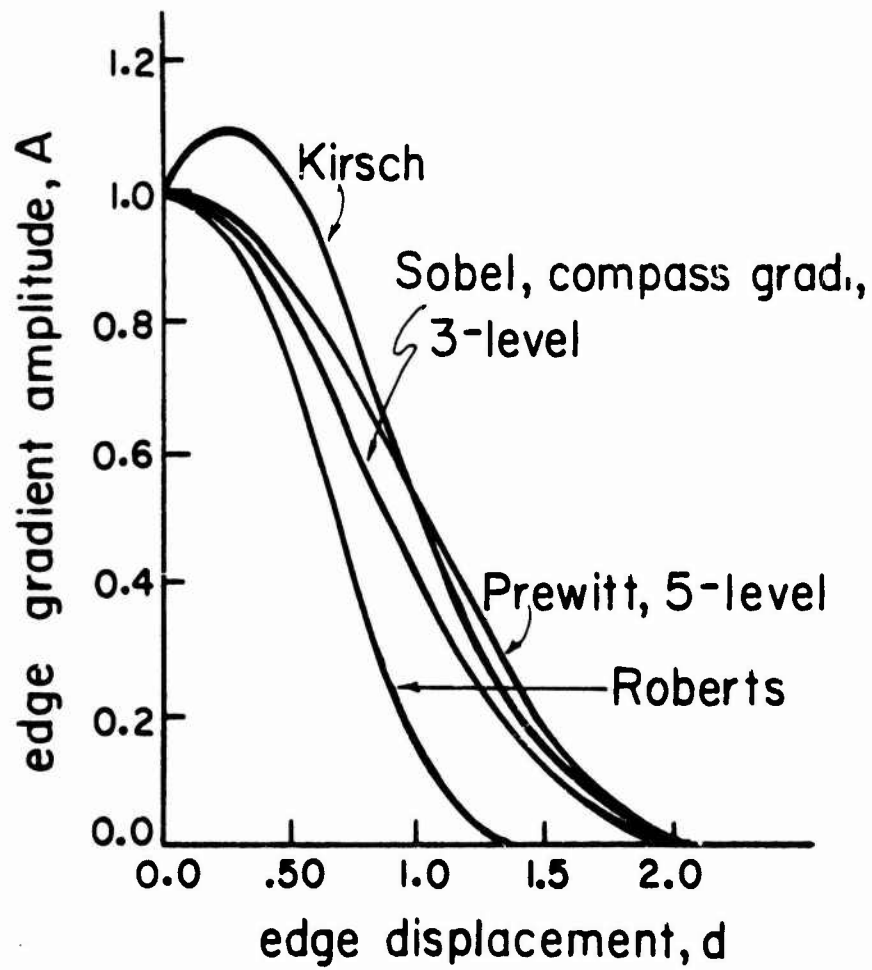
and for the Kirsch operator

$$A = \begin{cases} h[5+10(1-d^2)-(\frac{1}{\sqrt{2}}-d)^2] & 0 \leq d \leq \frac{1}{\sqrt{2}} \\ h[5-5(d-\frac{1}{\sqrt{2}})^2+2(\sqrt{2}-d)^2] & \frac{1}{\sqrt{2}} \leq d \leq \sqrt{2} \\ 5h(\frac{3}{\sqrt{2}}-d)^2 & \sqrt{2} \leq d \leq \frac{3}{\sqrt{2}} \end{cases} \quad (2.20)$$



a) vertical edge

Figure 2.10. Edge gradient amplitude response as a function of edge displacement for 2x2 and 3x3 operators



b) diagonal edge

Figure 2.10. (Continued)

Plots of A for the different operators are given in Figure 2.10. In these curves, A is normalized with respect to its value for a central edge. These curves can be used to determine edge detector resolution. It should be noticed that small size operators have better resolution. Also, for operators with the same mask size, the resolution is slightly dependent on the mask shape.

The results obtained in this section show that edge detector performance in the case of edges with general location and orientation can be approximately determined from their performance in the case of central edges with vertical or diagonal orientations. This last case is used as the ideal edge model in the following chapters.

2.3 Edge Fitting Method - Hueckel's Algorithm

In edge fitting, the image function $F(x,y)$ defined over a subregion B is compared with an ideal edge model $S_p(x,y)$, where p is the edge parameters vector. The difference between the actual and ideal models is function of p , and by changing these parameters the difference can be minimized. Edge acceptance is based on the value of the minimum difference. If it is less than a given threshold t , the image subregion is classified as an edge with the corresponding parameter p_{min} . Usually the mean square error is used to measure the difference between the ideal and actual edge. This error is given in the form

$$E_p = \int \int_D [F(x,y) - S_p(x,y)]^2 dx dy \quad (2.21)$$

Minimization of the error E_p can be obtained by an iterative procedure which is time consuming. However it is possible to introduce approximations of Eq. 2.21 such that its minimization can be achieved by simple analytic methods. This was the basic contribution of Hueckel in his papers published in 1971 and 1973. In the first paper, Hueckel used an orthogonal transformation to solve the problem of edge fitting [8]. Later, he extended his ideas to general edge-line fitting [14]. The Hueckel algorithm can be summarized as follows: A circular subregion of the image is compared with the edge model shown in Figure 2.11. The luminance function $S_p(x,y)$ of this edge-line model is given by

$$S_p = \begin{cases} b_- & \Delta \leq r_- \leq r_+ \\ b_- + t_- & r_- < \Delta \leq r_+ \\ b_- + t_- + t_+ & r_- \leq r_+ < \Delta \end{cases} \quad (2.22)$$

where

$$p = [c_x \quad c_y \quad r_- \quad r_+ \quad t_- \quad t_+ \quad b_-]^T \quad (2.23)$$

The functions $F(x,y)$ and $S_p(x,y)$ are expanded using a set of two dimensional orthogonal functions $\{H_i\}_0^\infty$. This set is chosen to be separable into the product of an angular and radial component. The error E_p is now in the form

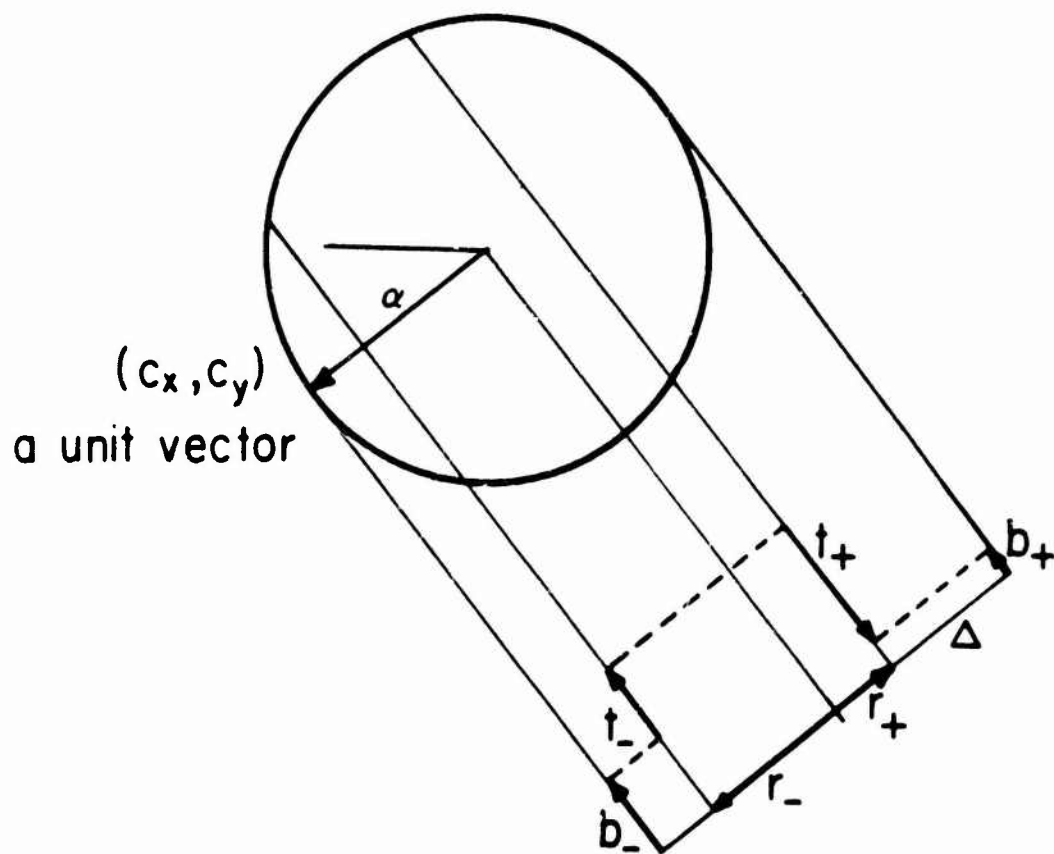


Figure 2.11. Hueckel's edge-line model

$$E_p = \sum_{i=0}^{\infty} (a_i - s_i)^2 \quad (2.24)$$

where

$$a_i = \iint_D H_i(x,y) F(x,y) dx dy \quad (2.25)$$

$$s_i = \iint_D H_i(x,y) S_p(x,y) dx dy \quad (2.26)$$

The series in Eq. 2.24 is approximated by its first nine components. The minimization of this truncated form and calculation of the corresponding p_{\min} can be achieved by solving simple algebraic equations. Hueckel argued that the truncation of the error series does not affect the performance of his algorithm because high frequency components are more related to image noise than to its signal contents.

The Hueckel algorithm has been considered by many as an almost optimum procedure for edge detection. A detailed analysis of this algorithm shows that this is not true. The basic difficulties with the Hueckel algorithm are the effect of the truncation of the series expansion and inaccuracies in the minimization procedure and computation of the edge parameters. These problems are discussed in Appendix A.

A major criticism of the previous approach to edge fitting is the fact that although images are usually discrete functions, the optimization procedure is derived in the continuous domain, thus the results obtained are suboptimum. This difficulty can be avoided by using the discrete image model in the derivation of the minimization procedure. An algorithm based on this idea will be introduced in Chapter 7.

2.4 Conclusion

In this chapter a review of some of the basic edge detection operators has been given. The operators chosen have the advantage of possessing simple mathematical formulas defined over a small region of the image, and thus it is not difficult to introduce a quantitative evaluation of their performance. In Chapters 3, 4, 5 and 6, different quantitative methods are used in the design and evaluation of the edge enhancement/thresholding operators. In Chapter 7, further investigation of the edge fitting technique is given.

Chapter 3

Statistical Model for Edge Detection

One of the methods which can be used in the evaluation of edge detection operators, is to test their performance in the case of an ideal signal with additive noise. This test is easy to implement. In addition, if the noise is assumed to be additive, white, and Gaussian, analytical results are not difficult to derive. Since edge detectors are used to classify different illumination inputs into edges or no edges, their performance can be tested by introducing inputs in the form of a noisy edge, or no edge, and then estimating the probability of making the right decision in each case. The following sections develop a statistical model for edge detection. Section 3.1 is a review of different decision rules used in hypothesis-testing. Section 3.2 evaluates the performance of the edge detectors for noisy edges. Section 3.3 discusses the estimation of the edge orientation.

3.1 Edge Detection as a Hypothesis-Testing Problem [4, 15, and 16]

In Section 2.1, the edge enhancement/thresholding

technique was described in detail. This technique closely resembles the hypothesis-testing algorithms used in classical statistical decision theory. The edge enhancement/thresholding operators have as an input an image subregion, with one of two hypotheses to be true,

H_1 : The subregion corresponds to an edge;

H_2 : The subregion corresponds to a no edge.

The edge detector calculates a function A of the input image, and accepts one of the two hypotheses according to the rule: Accept H_1 if

$$A \geq t \quad (3.1)$$

otherwise accept H_2 .

If the input image is noise free, it is possible to find a perfect decision strategy. On the other hand, if the image is affected by noise there will always be a possibility of making a wrong decision. For this case, four probabilities can be derived

$$P(\text{edge}|\text{edge}) = P(A \geq t | \text{edge}) \quad (3.2)$$

$$P(\text{no edge}|\text{no edge}) = P(A < t | \text{no edge}) \quad (3.3)$$

$$P(\text{no edge}|\text{edge}) = P(A < t | \text{edge}) \quad (3.4)$$

$$P(\text{edge}|\text{no edge}) = P(A \geq t | \text{no edge}) \quad (3.5)$$

The first two equations correspond to correct decisions,

while the other two correspond to incorrect decisions.

If the probabilities of occurrence of edges and no edges in a given image are known, then the probability of error will be in the form

$$P(\text{error}) = P(\text{no edge}|\text{edge})P(\text{edge}) + P(\text{edge}|\text{no edge}) \cdot P(\text{no edge}) \quad (3.6)$$

A decision procedure to minimize this probability of error is given by the rule: Decide an edge if

$$\frac{p(A|\text{edge})}{p(A|\text{no edge})} \geq \frac{P(\text{no edge})}{P(\text{edge})} \quad (3.7)$$

and decide no edge otherwise. This method is known as the Bayes decision rule for minimum probability of error. In Eq. 3.7, $p(A|\text{edge})$ and $p(A|\text{no edge})$ are the conditional probability density functions of A . A sketch of these probabilities is shown in Figure 3.1. The threshold t is set at a value which satisfies Eq. 3.7. In the special case, if edges and no edges are equally probable,

$$t = a \quad (3.8)$$

where a is the point of intersection of the two conditional probabilities.

If, in addition, the costs of taking one of the four decisions are known, namely $C(\text{edge}|\text{edge})$, . . . , $C(\text{no edge}|\text{no edge})$, then a decision procedure to minimize the average cost is to decide an edge if

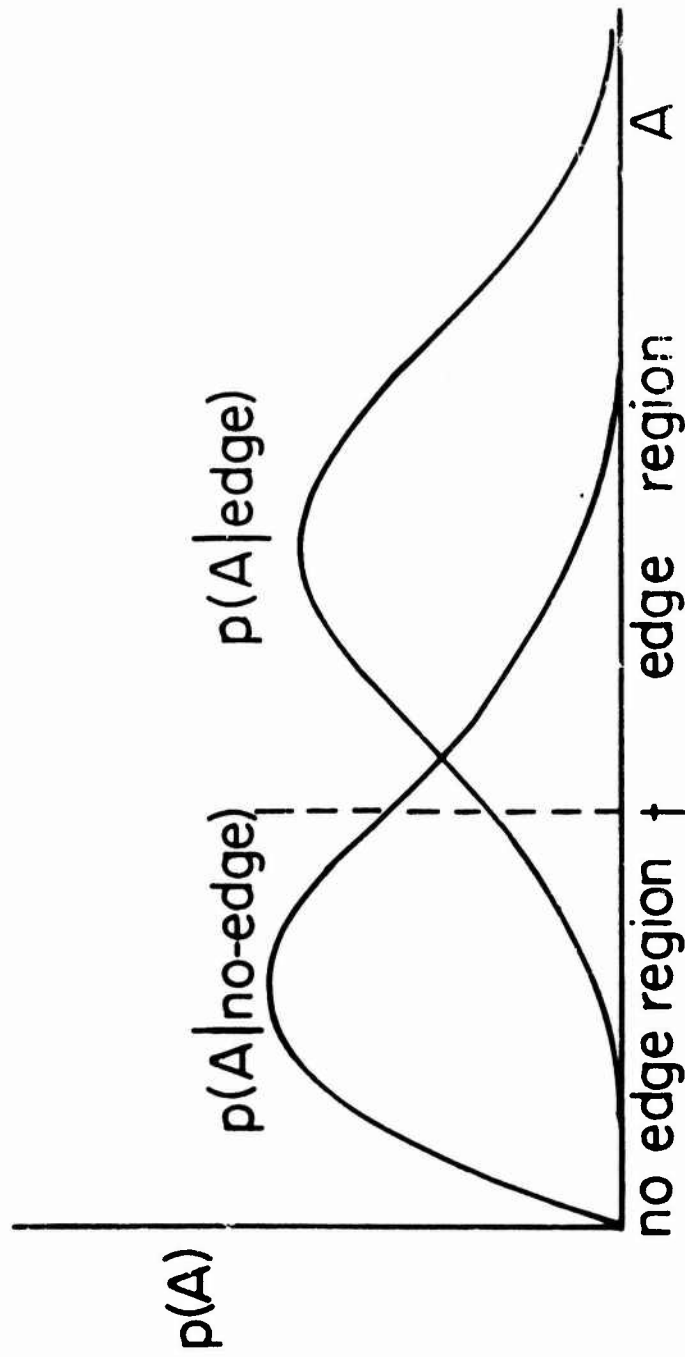


Figure 3.1. Typical conditional probability density functions of edge enhancement

$$\frac{p(A|\text{edge})}{p(A|\text{no edge})} \geq \frac{[C(\text{edge}|\text{no edge}) - C(\text{no edge}|\text{no edge})]}{[C(\text{no edge}|\text{edge}) - C(\text{edge}|\text{edge})]} \frac{P(\text{no edge})}{P(\text{edge})} \quad (3.9)$$

Otherwise, decide no edge. The threshold t can be specified accordingly.

In more general cases, when the probabilities of edges or no edges are not known. The threshold t can be set by one of the following two methods.

In the first method, t is set to achieve a given probability of missing an edge, $P(\text{no edge}|\text{edge})$, while minimizing the probability of false detection, $P(\text{edge}|\text{no edge})$. In this case, t is the solution of the equation

$$P(\text{no edge}|\text{edge}) = \int_{-\infty}^t p(A|\text{edge}) dA \quad (3.10)$$

This method, known as the Neyman-Pearson criterion, is frequently used in Radar detection.

In the second method, t is set to minimize the maximum possible error, that occurs when the probabilities of edges or no edges change for different input images. In this case the edge detector threshold is chosen such that

$$P(\text{edge}|\text{no edge}) = P(\text{no edge}|\text{edge}) \quad (3.11a)$$

or

$$\int_t^{\infty} p(A|\text{no edge})dA = \int_{-\infty}^t p(A|\text{edge})dA \quad (3.11b)$$

This is known as the minimax criterion.

Any of the previous decision strategies can be used in the design of edge detectors, especially the Neyman-Pearson criterion, which does not require the knowledge of the probabilities of edges or no edges. After choosing the threshold t , the performance of the edge detector can be evaluated as a function of the probabilities of detection and false detection. Computation of these probabilities for the edge enhancement/thresholding operators is given in the following section.

3.2 Edge Detector Performance, Case of Ideal Edge Plus Noise

In the model used in this section, an image subregion is considered to be the sum of two components. The first is an ideal central edge with orientations $\phi = 0$ or $\pi/4$, while the second is an additive white Gaussian noise with zero mean and standard deviation σ . The actual intensity f_j is then given by

$$f_j = s_j + n_j \quad (3.12)$$

where s_j and n_j are the ideal and noise components, respectively. The random variable f_j has the probability

density function

$$p(f_j) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left[-\frac{(f_j - s_j)^2}{2\sigma^2}\right] \quad (3.13)$$

When an edge detector is applied on this image model, the output of the i 'th convolving mask is given by

$$G_i = \sum_j M_i(j) f_j \quad (3.14)$$

where $M_i(j)$ are the components of the mask H_i . In this case $\{G_i\}$ will be joint Gaussian with the probability density function

$$p(\underline{G}) = (2\pi)^{-\frac{m}{2}} |\underline{\Sigma}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\underline{G} - \underline{\tilde{G}})^T \underline{\Sigma}^{-1} (\underline{G} - \underline{\tilde{G}})\right] \quad (3.15)$$

In Eq. 3.15, \underline{G} and $\underline{\tilde{G}}$ are vectors of the actual and ideal masks outputs given by

$$\underline{G} = [G_1 \quad G_2 \quad \dots \quad G_m]^T \quad (3.16)$$

$$\underline{\tilde{G}} = [\tilde{G}_1 \quad \tilde{G}_2 \quad \dots \quad \tilde{G}_m]^T \quad (3.17)$$

with

$$\tilde{G}_i = \sum_j M_i(j) s_j \quad (3.18)$$

Also, the covariance matrix $\underline{\Sigma}$ is given by

$$\underline{\Sigma} = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \cdot & \cdot & \cdot & \cdot & \cdot & \sigma_{1m}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \cdot & \cdot & \cdot & \cdot & \cdot & \sigma_{2m}^2 \\ & & \cdot & & & & & \\ & & & \cdot & & & & \\ \sigma_{m1}^2 & \sigma_{m2}^2 & & & \cdot & & & \sigma_{mm}^2 \end{bmatrix} \quad (3.19)$$

with

$$\sigma_{kl} = \sigma^2 \sum_j M_k(j) M_l(j) \quad (3.20)$$

The analysis introduced so far applies to both simple differential and template matching operators. To obtain expressions for the probability density function of A, each group of edge detectors has to be considered separately.

3.2.1 Simple Differential Operators

With the Roberts, Sobel, and Prewitt operators, two convolving masks are used. The outputs X and Y are joint Gaussian with mean and covariance matrix as given in Table 3.1.

From Table 3.1, it can be noticed that the random variables X and Y are independent. If the nonlinear function used is the square root, then

$$A = (X^2 + Y^2)^{\frac{1}{2}} \quad (3.21)$$

and the probability density function of A in the case of no edge is given by [17]. Thus,

TABLE 3.1

Mean Vector and Covariance Matrix of
Differential Gradient Operators

Operator	\tilde{G}			Σ
	no edge	vertical edge	diagonal edge	
Roberts	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -1 \end{bmatrix}_h$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}_h$	$\begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix}_\sigma$
Sobel	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 0 \end{bmatrix}_h$	$\begin{bmatrix} 3 \\ 3 \end{bmatrix}_h$	$\begin{bmatrix} \sqrt{12} & 0 \\ 0 & \sqrt{12} \end{bmatrix}_\sigma$
Prewitt	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 3 \\ 0 \end{bmatrix}_h$	$\begin{bmatrix} 2 \\ 2 \end{bmatrix}_h$	$\begin{bmatrix} \sqrt{6} & 0 \\ 0 & \sqrt{6} \end{bmatrix}_\sigma$

$$p(A) = \begin{cases} \frac{A}{\sigma_r^2} \exp\left[-\frac{A^2}{2\sigma_r^2}\right] & A \geq 0 \\ 0 & A < 0 \end{cases} \quad (3.22)$$

while in the case of an edge

$$p(A) = \begin{cases} \frac{A}{\sigma_r^2} \exp\left[-\frac{(A^2+a^2)}{2\sigma_r^2}\right] I_0\left(\frac{Aa}{\sigma_r^2}\right) & A \geq 0 \\ 0 & A < 0 \end{cases} \quad (3.23)$$

where σ_r is the diagonal elements of $\underline{\Sigma}$, and

$$a^2 = \tilde{X}^2 + \tilde{Y}^2 \quad (3.24)$$

In Eq. 3.23, $I_0(\cdot)$ is the modified Bessel function of zero order.

The previous probability density functions can be used to determine the probability of false detection P_F and the probability of correct detection P_D , for a given threshold t . These probabilities are of the form [18]

$$P_F = \exp\left(-\frac{t^2}{2\sigma_r^2}\right) \quad (3.25)$$

$$P_D = Q\left(\frac{a}{\sigma_r}, \frac{t}{\sigma_r}\right) \quad (3.26)$$

where $Q(a,b)$ is Marcum's Q-function defined as

$$Q(a,b) = \int_b^\infty x \exp\left[-\frac{a^2+x^2}{2}\right] I_0(ax) dx \quad (3.27)$$

If the nonlinear function used, is the sum of magnitudes

$$A = |X| + |Y| \quad (3.28)$$

the probability density function $p(A)$ can be derived in the form

$$p(A) = \frac{1}{2\sqrt{\pi}\sigma_r} \exp\left(-\frac{A^2+a^2}{4\sigma_r^2}\right) [p_1(\tilde{X}, \tilde{Y}) + p_1(\tilde{X}, -\tilde{Y}) + p_1(-\tilde{X}, \tilde{Y}) + p_1(-\tilde{X}, -\tilde{Y})] \quad (3.29)$$

where

$$p_1(X, Y) = \exp\left[\frac{(X-Y)A-XY}{2\sigma_r^2}\right] \left[\operatorname{erf}\left(\frac{A+X+Y}{\sqrt{2}\sigma_r}\right) + \operatorname{erf}\left(\frac{A-X-Y}{\sqrt{2}\sigma_r}\right) \right] \quad (3.30)$$

The corresponding probabilities P_F and P_D are

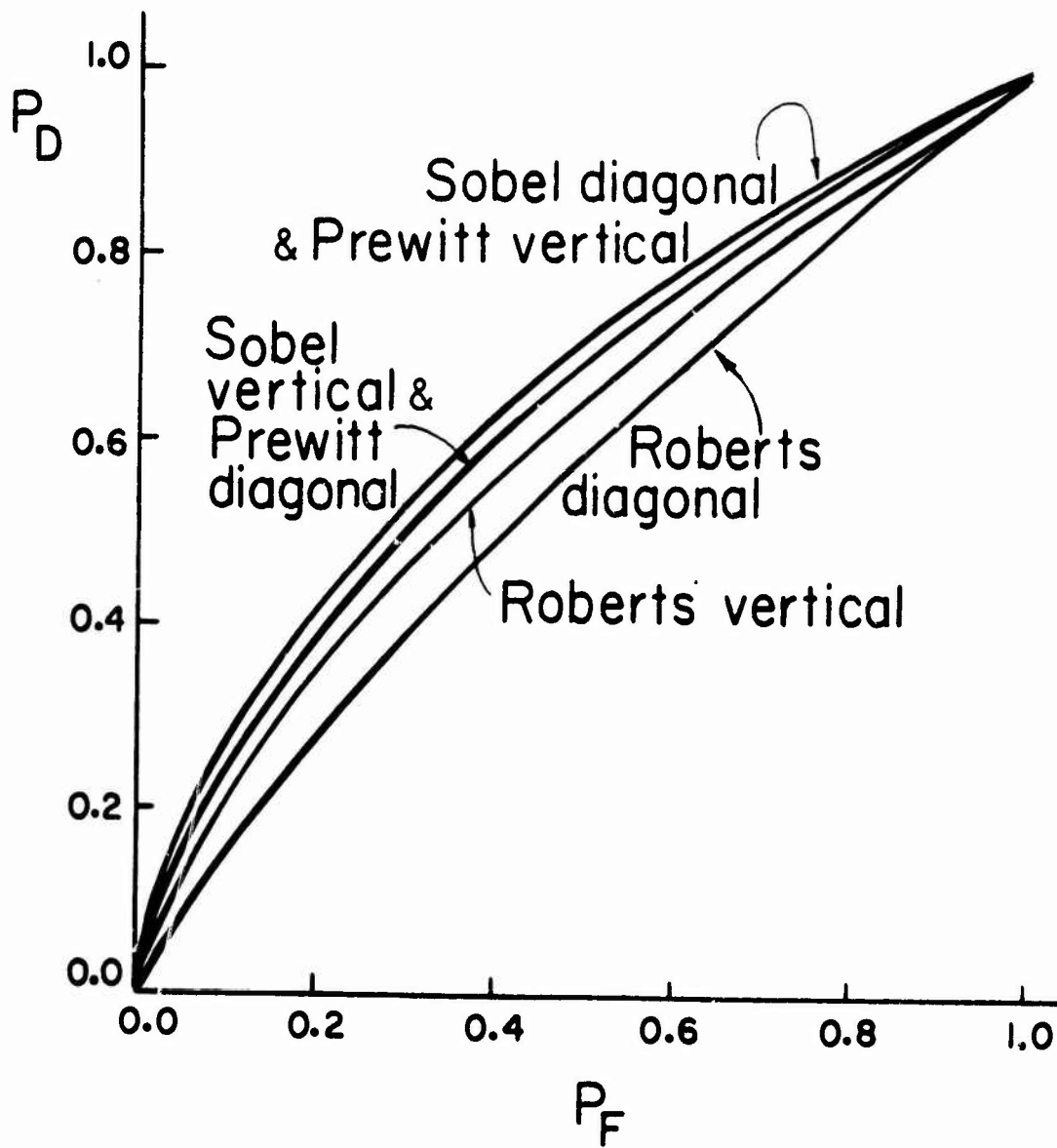
$$P_F = 1 - \left[2\operatorname{erf}\left(\frac{t}{\sqrt{2}\sigma_r}\right) \right]^2 \quad (3.31)$$

$$P_D = 1 - \left[\operatorname{erf}\left(\frac{t+\tilde{X}+\tilde{Y}}{\sqrt{2}\sigma_r}\right) + \operatorname{erf}\left(\frac{t-\tilde{X}-\tilde{Y}}{\sqrt{2}\sigma_r}\right) \right] \left[\operatorname{erf}\left(\frac{t+\tilde{X}-\tilde{Y}}{\sqrt{2}\sigma_r}\right) + \operatorname{erf}\left(\frac{t-\tilde{X}+\tilde{Y}}{\sqrt{2}\sigma_r}\right) \right] \quad (3.32)$$

In the previous equations

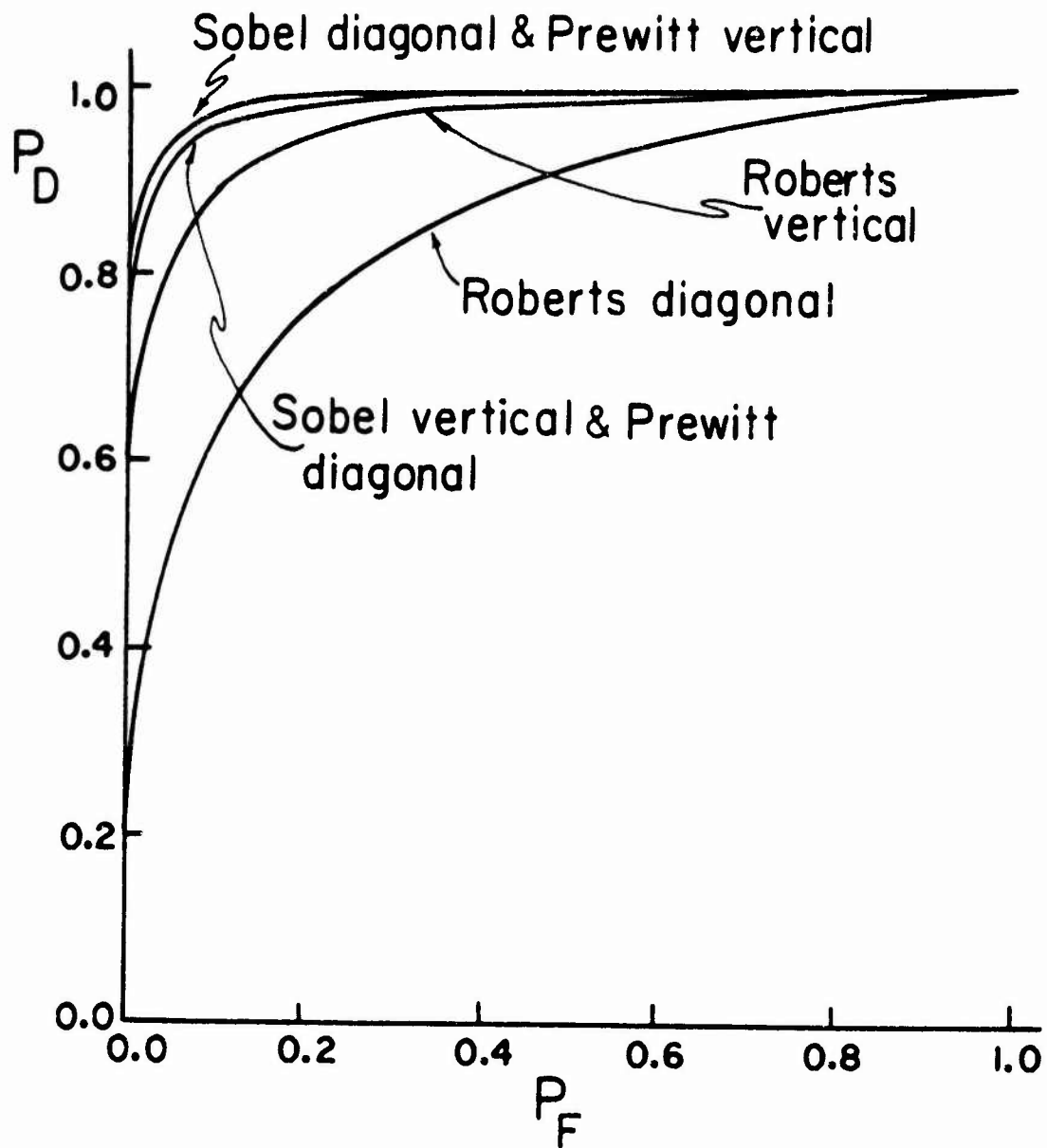
$$\operatorname{erf}(x) = \int_0^x \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^2}{2}\right) dy \quad (3.33)$$

To compare the performance of the Roberts, Sobel and Prewitt operators, the probability of correct detection P_D is plotted as a function of the probability of false detection P_F . Figure 3.2 presents such plots for vertical



a) SNR = 1.0

Figure 3.2. Probability of detection versus probability of false detection for simple differential operators



b) SNR = 10.0

Figure 3.2. (Continued)

and $\pi/4$ edges, with signal-to-noise ratios, $SNR^* = 1.0$ and 10.0. From these curves it is clear that the Sobel and Prewitt operators are superior to the Roberts operator. The prewitt operator is better than the Sobel operator for a vertical edge. But, for a diagonal edge, the Sobel operator is superior.

3.2.2 Template Matching Operators

With the compass gradient, Kirsch, 3-level, and 5-level operators, eight convolving masks are used. The output vector \underline{G} is a joint Gaussian with mean and covariance matrix as given in Table 3.2. The mean $\tilde{\underline{G}}$ is zero for no edge, and $\tilde{\underline{G}}$ for $\pi/4$ edge is the same as $\tilde{\underline{G}}$ for vertical edge with all the components shifted one position downward.

For these operators, computation of $p(A)$ is not straight forward. However, their performance can be evaluated using the probability density function $p(\underline{G})$. As an example

*The signal-to-noise ratio is defined as

$$SNR = \left(\frac{\text{edge height}}{\text{noise standard deviation}} \right)^2$$

TABLE 3.2

Mean Vector and Covariance Matrix of
Template Matching Operators for a Vertical Edge

OPERATOR	\bar{G}	Σ
3-Level	$\begin{bmatrix} 3 \\ 2 \\ 0 \\ -2 \\ -3 \\ -2 \\ 0 \\ 2 \end{bmatrix} h$	$\begin{bmatrix} 6 & 4 & 0 & -4 & -6 & -4 & 0 & 4 \\ 4 & 6 & 4 & 0 & -4 & -6 & -4 & 0 \\ 0 & & . & & & & . & \\ -4 & & & . & & & & . \\ -6 & & & & . & & & . \\ -4 & . & & & & . & & . \\ 0 & & . & & & & . & . \\ 4 & & & . & & & & . \end{bmatrix} \sigma^2$
5-Level	$\begin{bmatrix} 4 \\ 3 \\ 0 \\ -3 \\ -4 \\ -3 \\ 0 \\ 3 \end{bmatrix} h$	$\begin{bmatrix} 12 & 8 & 0 & -8 & -12 & -8 & 0 & 8 \\ 8 & 12 & 8 & 0 & -8 & -12 & -8 & 0 \\ 0 & & . & & & & . & \\ -8 & & & . & & & & . \\ -12 & & & & . & & & . \\ -8 & . & & & & . & & . \\ 0 & & . & & & & . & . \\ 8 & & & . & & & & . \end{bmatrix} \sigma^2$
Compass Gradient	$\begin{bmatrix} 3 \\ 2 \\ 0 \\ -2 \\ -3 \\ -2 \\ 0 \\ 2 \end{bmatrix} h$	$\begin{bmatrix} 12 & 8 & 4 & 0 & 0 & 0 & 4 & 8 \\ 8 & 12 & 8 & 4 & 0 & 0 & 0 & 4 \\ 4 & & . & & & & . & \\ 0 & & & . & & & & . \\ 0 & & & & . & & & . \\ 0 & . & & & & . & & . \\ 4 & & . & & & & . & . \\ 8 & & & . & & & & . \end{bmatrix} \sigma^2$
Kirsch	$\begin{bmatrix} 12 \\ 8 \\ 0 \\ -8 \\ -12 \\ -8 \\ 0 \\ 8 \end{bmatrix} h$	$\begin{bmatrix} 120 & 56 & -8 & -72 & -72 & -72 & -8 & 56 \\ 56 & 120 & 56 & -8 & -72 & -72 & -72 & -8 \\ -8 & & . & & & & . & \\ -72 & & & . & & & & . \\ -72 & & & & . & & & . \\ -72 & . & & & & . & & . \\ -8 & & . & & & & . & . \\ 8 & & & . & & & & . \end{bmatrix} \sigma^2$

$$\begin{aligned}
P_F &= P(A \geq t | \text{no edge}) \\
&= P(|G_1| \geq t \cdot \text{OR} \cdot |G_2| \geq t \dots \text{OR} \cdot |G_8| \geq t | \text{no edge}) \\
&= 1 - \int_{-t}^t \dots \int_{-t}^t p(\underline{G} | \text{no edge}) dG_1 dG_2 \dots dG_8 \quad (3.34)
\end{aligned}$$

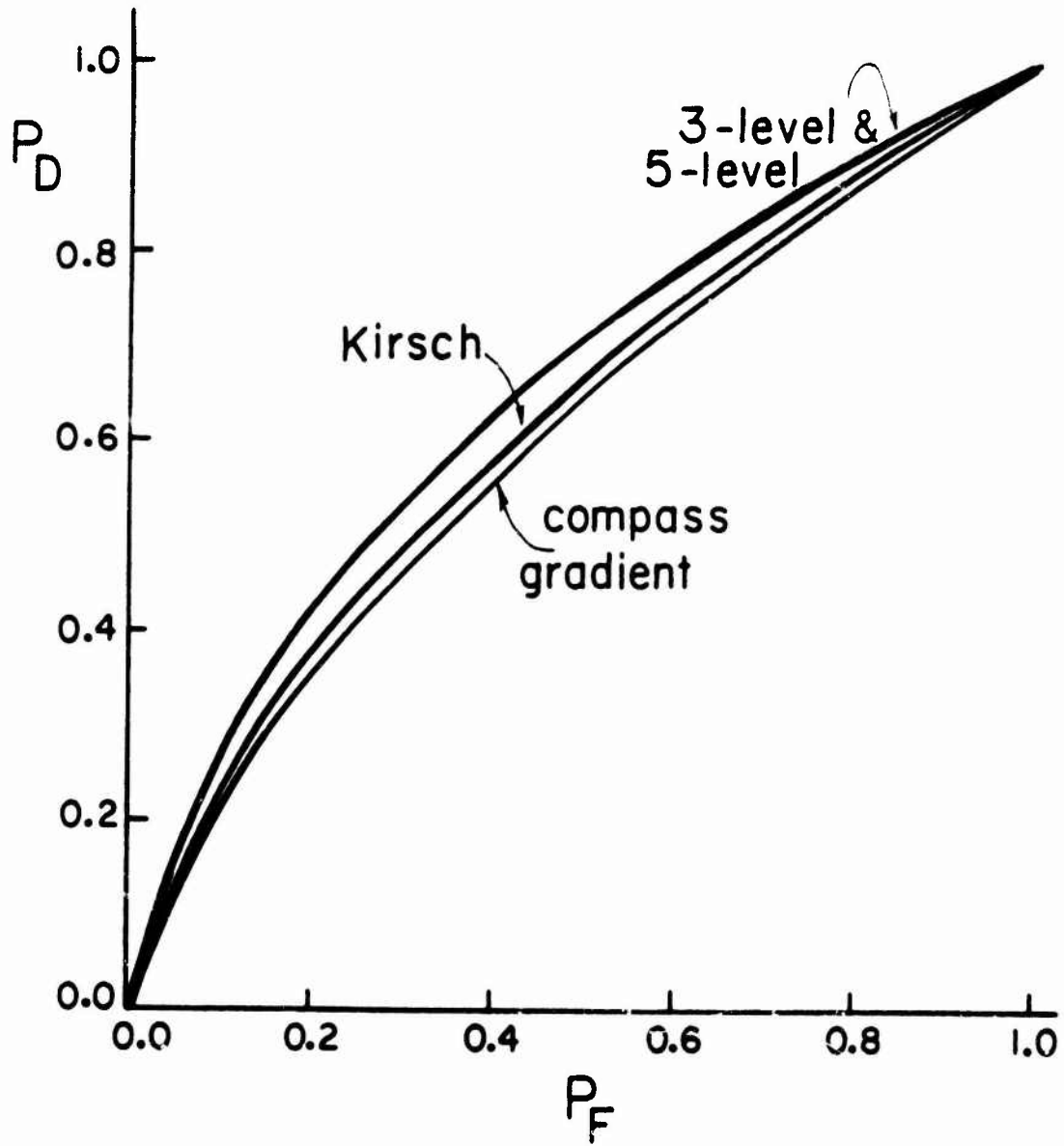
Equation 3.34 can be evaluated numerically using the parameters in Table 3.2. In Figure 3.3, P_D is plotted as a function of P_F for the different template matching operators for SNR = 1.0 and 10.0. From these curves, it is clear that the 3-level and 5-level operators have the best performances, followed by the Kirsch and finally the compass gradient operator. This can be explained by the fact that with the Kirsch and compass gradient operators more points are used in evaluating A, and thus, more noise is introduced, while these points are combined in such a way that they do not enhance the edge output.

3.3 Estimation of the Edge Orientation

The analysis in the previous section can be extended to the estimation of edge orientation. For the simple differential operators, the edge orientation is determined by the angle

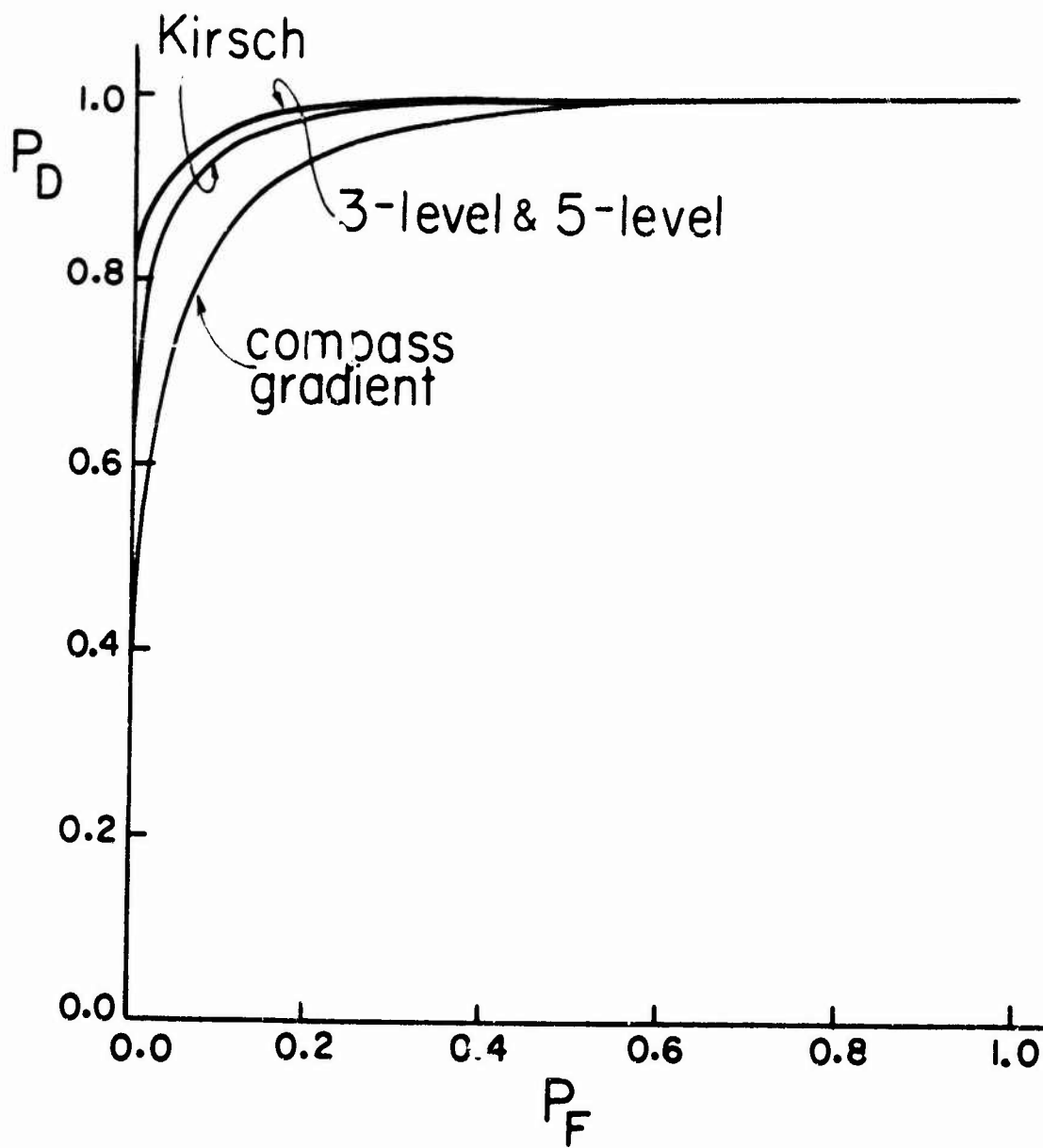
$$\theta_0 = \tan^{-1} \left(\frac{Y}{X} \right) \quad (3.35)$$

If X and Y correspond to no edge, they are zero mean Gaussian random variables. In this case, θ_0 is a random variable with $p(\theta_0)$ given by



a) SNR = 1.0

Figure 3.3. Probability of detection versus probability of false detection for template matching operators



b) SNR = 10.0

Figure 3.3. (Continued)

$$p(\theta_0) = \frac{1}{2\pi} \quad (3.36)$$

for $0 \leq \theta_0 \leq 2\pi$. If Y and X correspond to an edge, their means are non zero in general, and $p(\theta_0)$ is given by [19]

$$p(\theta_0) = \frac{e^{-a^2}}{2\pi} \left[1 + 2\sqrt{\pi} a \cos \gamma \left(\frac{1 + 2\operatorname{erf}\left(\frac{a \cos \gamma}{\sqrt{2}}\right)}{2} \right) \exp(a^2 \cos^2 \gamma) \right] \quad (3.37)$$

where

$$a = \left(\frac{X^2 + Y^2}{2\sigma_r^2} \right)^{\frac{1}{2}} \quad (3.38)$$

and

$$\gamma = \theta_0 - \tan^{-1} \frac{\tilde{Y}}{\tilde{X}} \quad (3.39)$$

The conditional probability of estimating the edge orientation, within a tolerance $\Delta\phi$, given that the region corresponds to an edge with orientation ϕ , is in the form

$$P(\phi - \Delta\phi \leq \theta \leq \phi + \Delta\phi | \text{edge}, \phi) = \int_{\phi - \Delta\phi}^{\phi + \Delta\phi} p(\theta | \text{edge}, \phi) d\theta \quad (3.40)$$

it should be noticed that the probability of the exact estimation of the orientation of a noisy edge is zero.

For the template matching operators, the detection of the edge orientation angle can be considered as multiple-hypotheses testing. If the actual edge angle is θ , the probability of making a correct decision is

$$\begin{aligned}
P(\theta = \theta_i | \text{edge}, < \theta_i) &= P(G_i > G_k \forall k | \text{edge}, < \theta_i) \\
&= \int_{G_i=t}^{\infty} \int_{G_k=-\infty}^{G_i} \dots \int_{k \neq i} p(\underline{G} | \text{edge}, < \theta_i) dG_1 dG_2 \dots dG_8
\end{aligned}
\tag{3.41}$$

Equation 3.41 can be evaluated numerically.

Since the estimation of the edge orientation is affected by more sources of error, compared with the detection of the edge presence or absence, this additional information should be used carefully. An unwise usage of the estimated edge orientation may reduce edge detector performance. More research is needed to find an optimum strategy for using edge orientation information.

3.4 Conclusion

In this chapter, a statistical model for edge detection has been developed. The performance of the different edge detectors is evaluated for actual central edges with specific edge orientations. The success in introducing such a model helps in transferring the communication theory concepts into edge detection problems. This is a major point in the analysis and design of edge detectors, because many problems in edge detection have already been solved in communication theory. It is interesting to notice that the magnitude and angle of the

simple differential operators have the same probability density functions of the envelope and phase of narrowband signal with additive Gaussian noise [19]. Other examples can be noticed and used successfully.

Chapter 4

Edge Detection as a Pattern Classification Problem

Edge detection as a hypothesis-testing problem was presented in Chapter 3. Another approach, which is introduced in this chapter, is to consider edge detection as a pattern classification problem. The edge detector has as its input different image subregions, and it is required to classify these subregions into the class of edges Ω_1 , and the class of no edges Ω_2 . The decision strategy given by Eq. 2.3 can be written in the form

$$\text{If } w(1)A + w(2) > 0 \quad \text{then } A \in \Omega_1 \quad (4.1a)$$

$$\text{and if } w(1)A + w(2) < 0 \quad \text{then } A \in \Omega_2 \quad (4.1b)$$

where the weighting vector $\underline{w} = [w(1) \ w(2)]^T$ is related to the threshold t by the relation

$$t = - \frac{w(2)}{w(1)} \quad (4.2)$$

The components of \underline{w} are obtained by training the edge detector using a set of known edge and no edge patterns. After this training phase, the edge detector is used to classify unknown prototypes in actual images. The

performance with actual images will depend on the procedure used in the training phase. There are different methods that can be used in training a pattern classifier. A review of these methods is given in Section 4.1. One of these methods, the Ho-Kashyap algorithm, will be used in the edge detectors design. The basic concepts of this algorithm and the reason behind its choice are discussed in Section 4.2. Experimental results are summarized in Section 4.3.

4.1 Training Methods for Pattern Classifiers

The decision function in Eq. 4.1 is based on the scalar variable A . This decision function can be generalized to the n -dimensional case

$$d(\underline{x}_n) = \underline{w}_n^T \underline{x}_n + w(n+1) \quad (4.3)$$

where $\underline{x}_n = [x(1), x(2), \dots, x(n)]^T$ is the pattern vector and $\underline{w}_n = [w(1), w(2), \dots, w(n)]^T$ is the weight vector. Usually, Eq. 4.3 is expressed in the form

$$d(\underline{x}) = \underline{w}^T \underline{x} \quad (4.4)$$

where $\underline{x} = [x(1), x(2), \dots, x(n), 1]^T$ is an augmented pattern vector and $\underline{w} = [w(1), w(2), \dots, w(n), w(n+1)]^T$ is an augmented weight vector, [20]. The decision strategy is then

$$\text{If } \underline{w}^T \underline{x} > 0 \implies \underline{x} \in \Omega_1 \quad (4.5a)$$

$$\text{and if } \underline{w}^T \underline{x} < 0 \implies \underline{x} \in \Omega_2 \quad (4.5b)$$

In the training phase, the pattern classifier is given two sets of prototype patterns $\{\underline{x}_1, \underline{x}_2, \dots, \underline{x}_N\} \in \Omega_1$, and $\{\underline{x}_{N+1}, \underline{x}_{N+2}, \dots, \underline{x}_{2N}\} \in \Omega_2$. The weight vector \underline{w} is determined such that $\underline{w}^T \underline{x} > 0$ for all patterns of Ω_1 , and $\underline{w}^T \underline{x} < 0$ for all patterns of Ω_2 . If the patterns of Ω_2 are multiplied by (-1) , the required condition becomes $\underline{w}^T \underline{x} > 0$ for all patterns. The pattern classification problem is then reduced to finding a vector \underline{w} such that

$$\underline{X} \underline{w} > 0 \quad (4.6)$$

is satisfied, where

$$\underline{X} = \begin{bmatrix} \underline{x}_1^T \\ \underline{x}_2^T \\ \vdots \\ \underline{x}_{2N}^T \end{bmatrix} \quad (4.7)$$

if there exists a \underline{w} which satisfies Eq. 4.6, the classes are said to be separable; otherwise they are nonseparable [20].

One approach to the solution of the set of linear inequalities of Eq. 4.6 is to define a criterion function $J(\underline{w})$ that becomes minimum if \underline{w} satisfies Eq. 4.6. This reduces the problem to one of minimizing a scalar function;

a problem that can be solved by a gradient descent procedure [4]. An example of a criterion function, that can be used, is the perceptron criterion function

$$J_p(\underline{w}) = \sum_{\underline{x} \in \mathcal{X}} (-\underline{w}^T \underline{x}) \quad (4.8)$$

where \mathcal{X} is the set of samples misclassified by \underline{w} . Another example is

$$J_r(\underline{w}) = \frac{1}{2} \sum_{\underline{x} \in \mathcal{X}} \frac{(\underline{w}^T \underline{x} - b)^2}{\|\underline{x}\|^2} \quad (4.9)$$

where now \mathcal{X} is the set of samples for which $\underline{w}^T \underline{x} \leq b$. The previous two criterion functions focus their attention on the misclassified samples. A different criterion function that involves all the samples is

$$J_s(\underline{w}) = \|\underline{X} \underline{w} - \underline{b}\|^2 \quad (4.10)$$

where the components of \underline{b} are all positive. The minimization of $J_s(\underline{w})$ depends on the value of \underline{b} . If \underline{b} is fixed arbitrarily there is no guarantee that the solution will give a separating vector in the linearly separable case. To avoid that, \underline{b} and \underline{w} are allowed to vary in the minimization procedure. This is the basic concept of the Ho-Kashyap algorithm. Another approach to solve the inequalities in Eq. 4.6 is to use linear programming procedures. Details of these procedures and analysis of the other previous methods are given in references [4, 20].

In order to use any of the previous methods in the design of edge detectors, two conditions for the resulting vector \underline{w} are required. First, if the training patterns are separable, the training procedure should converge to a \underline{w} which classifies the patterns correctly. Second, if the training patterns are not separable, a case which is usually encountered in edge detection, the training procedure should detect the nonseparability and yield a solution which can be used practically. These two conditions are achieved only by the Ho-Kashyap algorithm [21], and by a linear programming procedure that minimizes the perceptron criterion function [22]. Any of these two methods can be used in edge detector design. The performance of each method will depend on the distribution of the classes. A comparison between the two methods is outside the scope of this dissertation. Therefore, in the following section a discussion of one of them, the Ho-Kashyap algorithm, and its application in edge detection, is given. A similar analysis can be developed for the linear programming procedure.

4.2 The Ho-Kashyap Algorithm

In this algorithm, the solution of the inequalities in Eq. 4.6 has been reformulated as a problem of finding \underline{w} and $\underline{b} > 0$ such that $J_g(\underline{w})$ in Eq. 4.10 is minimized. The minimizations can be achieved by a steepest descent

procedure that implements the gradient functions

$$\frac{\partial J_S}{\partial \underline{w}} = \underline{X}^T (\underline{X} \underline{w} - \underline{b}) \quad (4.11a)$$

and

$$\frac{\partial J_S}{\partial \underline{b}} = \underline{b} - \underline{X} \underline{w} \quad (4.11b)$$

Since there is no constraint on \underline{w} , $\frac{\partial J_S}{\partial \underline{w}} = 0$ implies

$$\begin{aligned} \underline{w} &= (\underline{X}^T \underline{X})^{-1} \underline{X}^T \underline{b} \\ &= \underline{X}^\# \underline{b} \end{aligned} \quad (4.12)$$

where $\underline{X}^\#$ is the pseudoinverse of \underline{X} . Since all the components of \underline{b} are constrained to be positive, this vector must be varied in such a manner to never violate this constraint. This can be accomplished by letting

$$\underline{b}(k+1) = \underline{b}(k) + \delta \underline{b}(k) \quad (4.13)$$

where

$$\delta \underline{b}(k) = c [\underline{e}(k) + |\underline{e}(k)|] \quad (4.14a)$$

and

$$\underline{e}(k) = \underline{X} \underline{w}(k) - \underline{b}(k) \quad (4.14b)$$

In Eqs. 4.13 and 4.14, k denotes the iteration index, c is a positive correction increment, and $|\underline{e}(k)|$ indicates the absolute value of each component of the error vector $\underline{e}(k)$ [20]. From Eqs. 4.12 and 4.13,

$$\underline{w}(k+1) = \underline{w}(k) + X^\# \delta \underline{b}(k) \quad (4.15)$$

Thus, Eq. 4.10 can be minimized through the iteration

$$\underline{w}(1) = X^\# \underline{b}(1) \quad (4.16)$$

$$\underline{e}(k) = X \underline{w}(k) - \underline{b}(k) \quad (4.17)$$

$$\underline{w}(k+1) = \underline{w}(k) + c X^\# [\underline{e}(k) + |\underline{e}(k)|] \quad (4.18)$$

$$\underline{b}(k+1) = \underline{b}(k) + c [\underline{e}(k) + |\underline{e}(k)|] \quad (4.19)$$

where $\underline{b}(1) > 0$ but otherwise is arbitrary, and c is a constant such that $0 < c \leq 1$.

If the patterns are separable, Eqs. 4.17 to 4.19 can be repeated until all components of $\underline{e}(k)$ converge to zero, or to any reasonably small value. On the other hand, if the components of $\underline{e}(k)$ cease to be positive, but are not all zero, at any iteration step, this will indicate that the classes are not separable [20, 21]. These two characteristics of the Ho-Kashyap algorithm are important, especially when the algorithm is used to design edge detectors. Because the degree of separability of the classes of edges and no edges changes for different image models, the procedure used in the edge detector design should be able to handle both separable and nonseparable patterns.

4.3 Application of the Ho-Kashyap Algorithm to Edge Detection

The Ho-Kashyap algorithm is used in the design of edge enhancement/thresholding operators. In this experiment, patterns of vertical edges, and patterns of no edges, are generated. Gaussian noise is added to produce edge prototypes with SNR = 1.0 or 10.0. The outputs of the different edge detectors in the case of a vertical edge $\{A_1, A_2, \dots, A_N\}$, and in the case of no edge $\{A_{N+1}, A_{N+2}, \dots, A_{2N}\}$, are used to construct the augmented matrix

$$\underline{X} = \begin{bmatrix} A_1 & 1 \\ \vdots & \vdots \\ A_N & 1 \\ -A_{N+1} & -1 \\ \vdots & \vdots \\ -A_{2N} & -1 \end{bmatrix} \quad (4.20)$$

The number of patterns of each class is chosen to be $N = 20$. This ensures that the performance on design and test data will be similar [4]. The initial components of $\underline{b}(1)$ are chosen to be unity, and iteration given by Eqs. 4.17 to 4.19 is repeated up to 500 times. The experiment is ended if the components of $\underline{e}(k)$ are all less than a small value, (0.001), or if nonseparability is proved. It is sometimes useful to end the iteration when the threshold $t = -w(2)/w(1)$ stabilizes within a relatively small variation.

After the training phase is finished, the values of w obtained are tested with a new set of 250 prototypes generated with the same model. The probability of detection in the case of an edge, and the probability of false detection in the case of a no edge, are calculated. The results obtained are compared with the theoretical results derived in Chapter 3. These results are given in Table 4.1 for different edge detectors with vertical and $\pi/4$ edges and SNR = 1.0 and 10.0, respectively. It should be noticed that in many cases the edge detector threshold t converges to a value which results in equal probabilities of error

$$P_F \approx 1 - P_D \quad (4.21)$$

This satisfies the Bayes minimum error criterion if edges and no edges are equally probable. Thus, the results obtained with the Ho-Kashyap algorithm have practical significance.

4.4 Conclusion

In this chapter, it has been shown that edge detectors can be designed using pattern classification techniques. As an example, the Ho-Kashyap algorithm, was used to design different edge enhancement/thresholding operators. The edge model used was an ideal edge plus Gaussian noise. This model helps in comparing the experimental results with

TABLE 4.1

Threshold Level and Error Probabilities for Ho-Kashyap Design Procedure

OPERATOR	Vertical Edge, SNR=1						Vertical Edge, SNR=10						Diagonal Edge, SNR=1						Diagonal Edge, SNR=10						
	experiment			theory			experiment			theory			experiment			theory			experiment			theory			
	T	P _D	P _F	P _D	P _F	P _E	T	P _D	P _F	P _D	P _F	P _E	T	P _D	P _F	P _D	P _F	P _E	T	P _D	P _F	P _D	P _F	P _E	
Roberts	1.36	52.0	37.6	55.89	39.87	0.67	91.2	11.6	89.16	10.47	1.74	55.60	46.80	55.10	46.88	0.78	74.80	20.80	77.79	22.11					
Square Root																									
Roberts	1.22	52.0	38.4	55.16	39.30	0.62	90.8	7.6	89.20	9.5	2.24	54.40	46.40	53.85	45.70	0.97	75.60	19.60	76.81	23.30					
Magnitude																									
Sobel	1.18	60.8	11.2	60.01	39.54	0.66	92.0	9.6	92.34	5.69	1.14	63.20	39.20	60.40	37.63	0.63	90.80	9.20	94.65	5.28					
Square Root																									
Prewitt	1.16	59.5	36.6	60.80	38.40	0.66	93.0	3.8	91.20	4.80	1.19	61.20	39.60	59.27	38.71	0.64	90.00	8.40	93.07	6.42					
Square Root																									
Compass	1.52	57.6	45.2	61.27	46.56	0.73	85.2	12.8	88.58	13.55	1.51	57.60	46.80	61.80	47.20	0.71	80.80	14.00	90.00	15.30					
Gradient																									
Kirsch	1.43	56.0	38.4	53.08	34.08	0.69	89.2	9.2	89.78	5.76	1.45	54.40	36.00	52.40	32.40	0.79	82.80	3.60	82.50	2.30					
3-Level	1.16	60.8	38.4	59.02	36.92	0.65	89.6	6.4	92.64	3.79	1.16	59.20	38.40	58.70	36.50	0.61	89.60	8.40	94.60	5.60					
5-Level	1.24	58.0	37.6	58.12	36.09	0.66	90.8	6.8	92.45	4.90	1.22	60.40	39.20	59.30	37.40	0.65	90.00	8.40	93.10	5.40					

the theoretical ones obtained in Chapter 3. The same technique can be easily extended to the design of any edge detector with any arbitrary noise model.

Chapter 5

Figure of Merit Comparison of Edge Detectors

The methods introduced in the previous two chapters can be used in both the evaluation and the design of edge detectors. In this chapter, a third method which can be used only in the evaluation of edge detectors performance, is introduced. The procedure used in this chapter can be summarized as follows. First, an artificial test image is generated. Second, an edge detector is applied on this test image. Third, the quality of the resulting edge map is measured in terms of a scalar function. That function can be considered as a figure of merit of the corresponding edge detector. The figure of merit used should be sensitive to the different expected errors so that it is maximum when the edge map is perfect, and decreases as the error in the edge map increases. Methods based on the previous technique have been introduced by Fram and Deutsch [13], and by Pratt [1]. This latter method has two advantages: it weights the different errors according to their importance; and it allows each edge detector to be tuned to its best capabilities, which guarantees a fair comparison. Because of these advantages, the experiments

discussed in the following sections will be based on the figure of merit developed by Pratt. Section 5.1 explains the basic ideas of this technique. Section 5.2 summarizes the results obtained with simple test images. Section 5.3 introduces conclusions based on the results of Chapters 3,4 and 5.

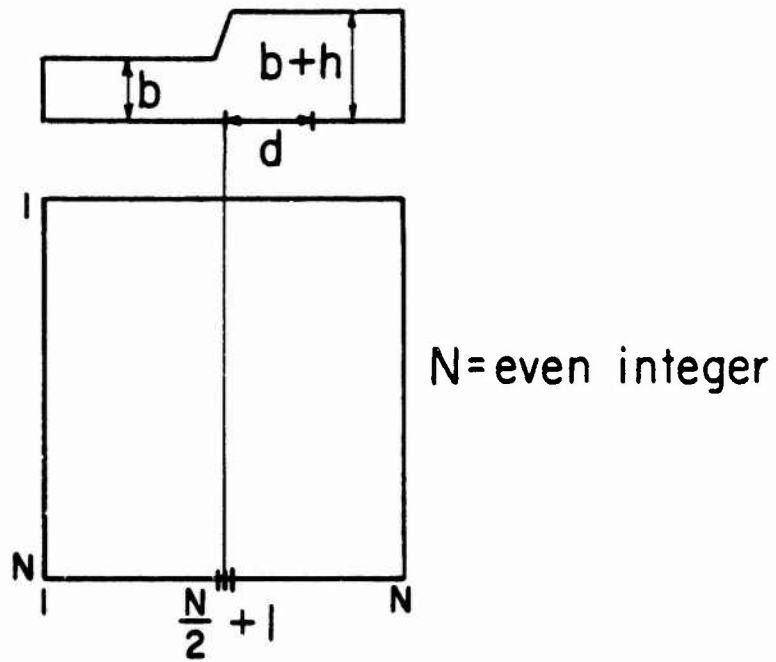
5.1 Figure of Merit Concepts

The procedure introduced by Pratt utilizes a test image consisting of a 64 x 64 pixels array over a 0 to 255 amplitude range with a vertically oriented edge of variable contrast and slope placed at its center. Independent Gaussian noise of standard deviation σ is added to the edge image, and the resultant picture is clipped to the maximum display limits. As in the previous chapters, the signal-to-noise ratio is defined as

$$\text{SNR} = \left(\frac{h}{\sigma} \right)^2 \quad (5.1)$$

where h is the edge height.

When an edge detector is applied on this test image, three major types of error will affect the resulting edge map: (a), missing of valid edge point; (b), failure to localize edge points; (c), classification of noise pulses as edge points. Examples of these errors are shown in Figure 5.1.



a) vertical edge test image



b) ideal



c) fragmented



d) offset



e) smeared

Figure 5.1. Types of edge detection errors

The quality of the resulting edge map may be assessed by the figure of merit defined by

$$F = \frac{1}{\max\{I_I, I_A\}} \sum_{i=1}^{I_A} \frac{1}{1+\alpha d^2} \quad (5.2)$$

where I_I and I_A represent the number of ideal and actual edge map points, respectively, α is a scaling constant, and d is the separation distance of an actual edge point normal to a line of ideal edge points. The rating factor is normalized so that $F = 1$ for a perfectly detected edge. The scaling factor α may be adjusted to penalize edges which are localized but offset from the true position. Normalization by the maximum of the actual and ideal number of edge points insures a penalty for smeared or fragmented edges. This figure of merit gives higher rating for a smeared edge than for an offset edge. This is reasonable because it is possible to thin the smeared edge by post-processing [1].

The figure of merit method has been used to evaluate the performance of the Roberts, Kirsch, Sobel, and compass gradient-operators. In each case, the thresholds are chosen to maximize the figure of merit, plots of these maximum values are given in [1]. The results obtained in this experiment can be predicted theoretically using the probabilities of detection of central edges P_D , of detection of displaced edges P_{Dis} , and of false detection

P_F , for a given edge detector. As an example, if a 3×3 edge detector is applied to the test image shown in Figure 5.1, there will be a central edge at column $\frac{N}{2} + 1$, displaced edges at the two adjacent columns, and no true edges elsewhere. For this case, Eq. 5.2 reduces to

$$F = \frac{N}{I_N} \left[P_D + \frac{2P_{Dis}}{1+\alpha} + P_F \left(\sum_{d=-2}^{\frac{N}{2}-1} \frac{1}{1+\alpha d^2} + \sum_{d=2}^{\frac{N}{2}-1} \frac{1}{1+\alpha d^2} \right) \right] \quad (5.3)$$

where

$$I_N = \max\{N, [P_D + 2P_{Dis} + (N-3)P_F]N\} \quad (5.4)$$

The analysis introduced thus far is based on a test image that contains a vertical edge. The same analysis can be extended to other image models, but in these cases the evaluation of Eq. 5.2 will become more difficult. Another test image which is relatively easy to analyze is one that contains a diagonal edge. As has been shown in Chapter 2, the results obtained from the vertical and the diagonal edge models are sufficient to determine edge detector performance.

A test image that contains a diagonal edge is shown in Figure 5.2. The image consists of 128×128 pixels generated with the same signal and noise models used in the test image that contains the vertical edge. To simplify the comparison of the results obtained in both cases, only

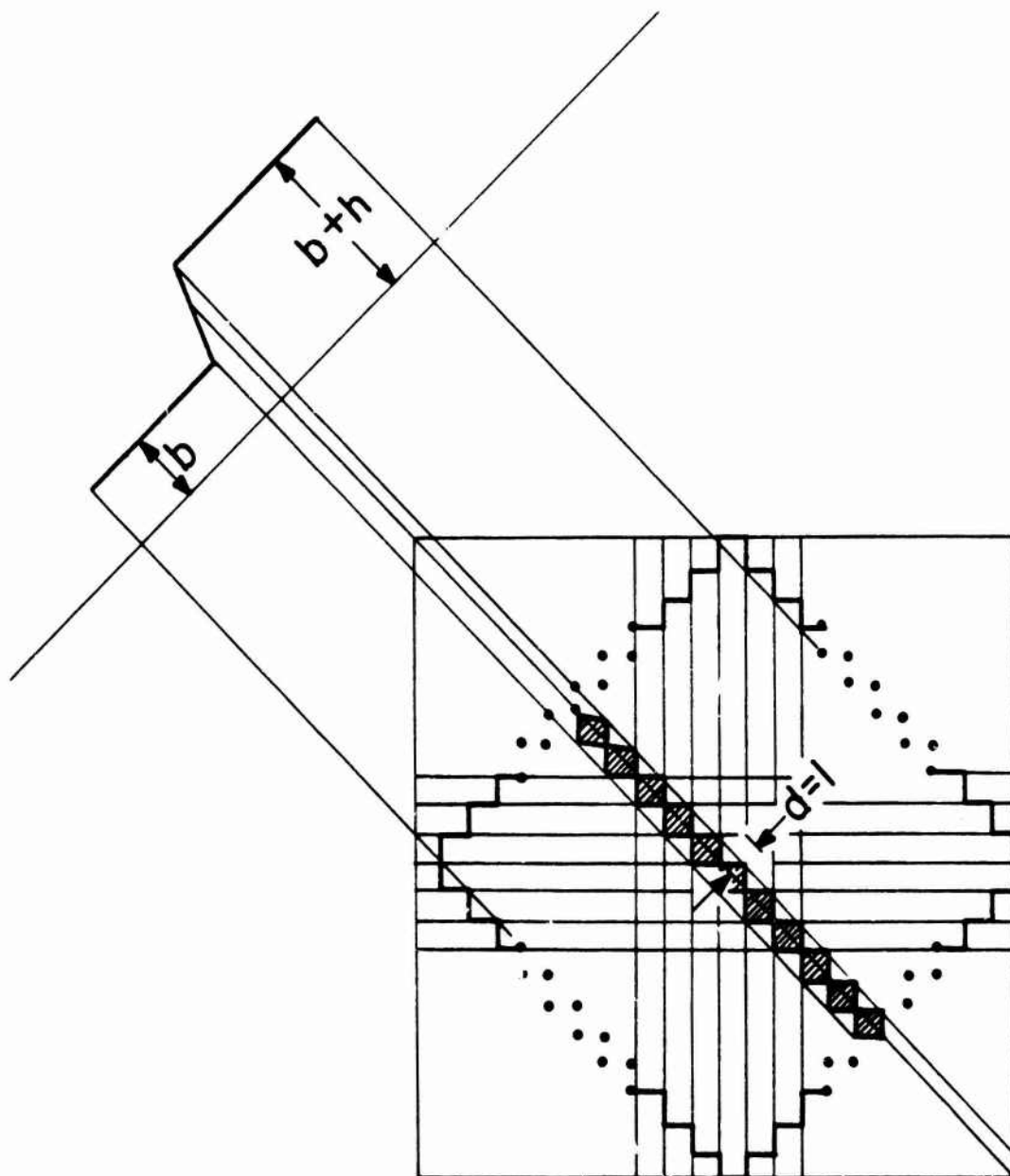
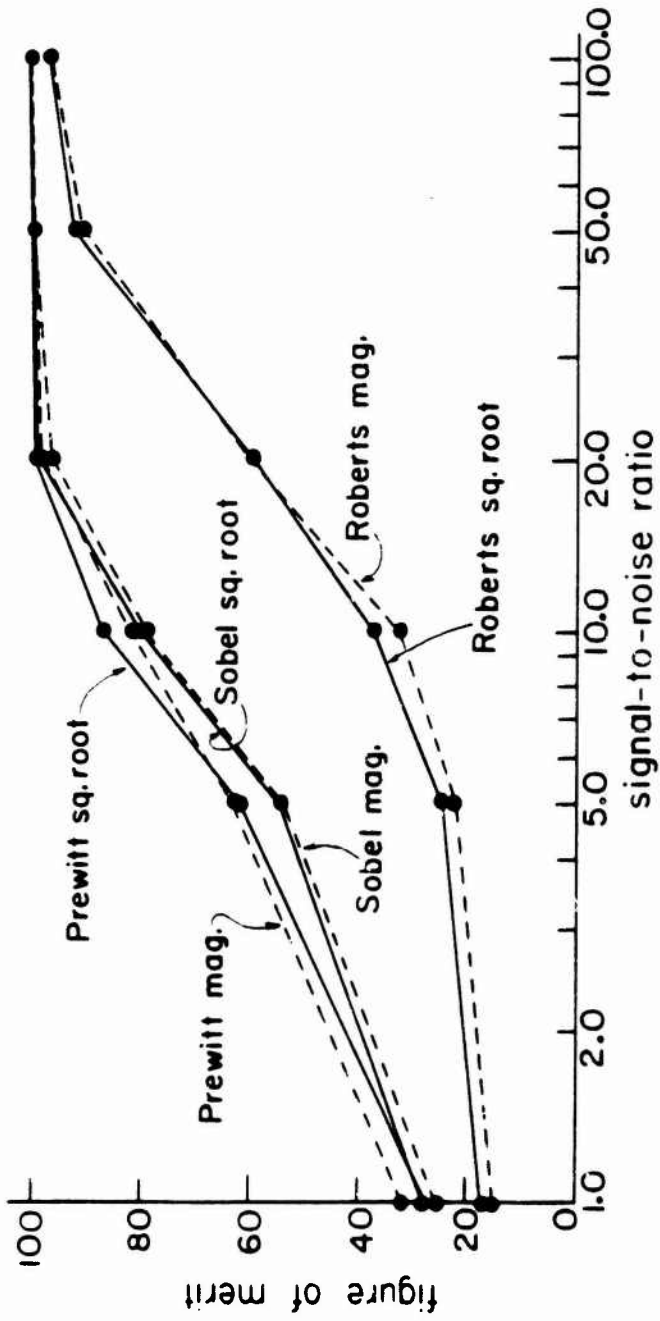


Figure 5.2. Figure of merit test image geometry for diagonal edge

the central part of the diagonal edge is used in calculating the figure of merit. This central region is shown bounded by dotted lines in Figure 5.2. The number of edge pixels in this region is chosen to be equal to the number of edge pixels in the vertical edge model. But, the number of non-edge pixels in the diagonal edge model is twice their number in the vertical edge model. The effect of this difference is compensated by scaling the diagonal distance d by a factor $\sqrt{2}$. The results obtained with these two test images will be given in the following section.

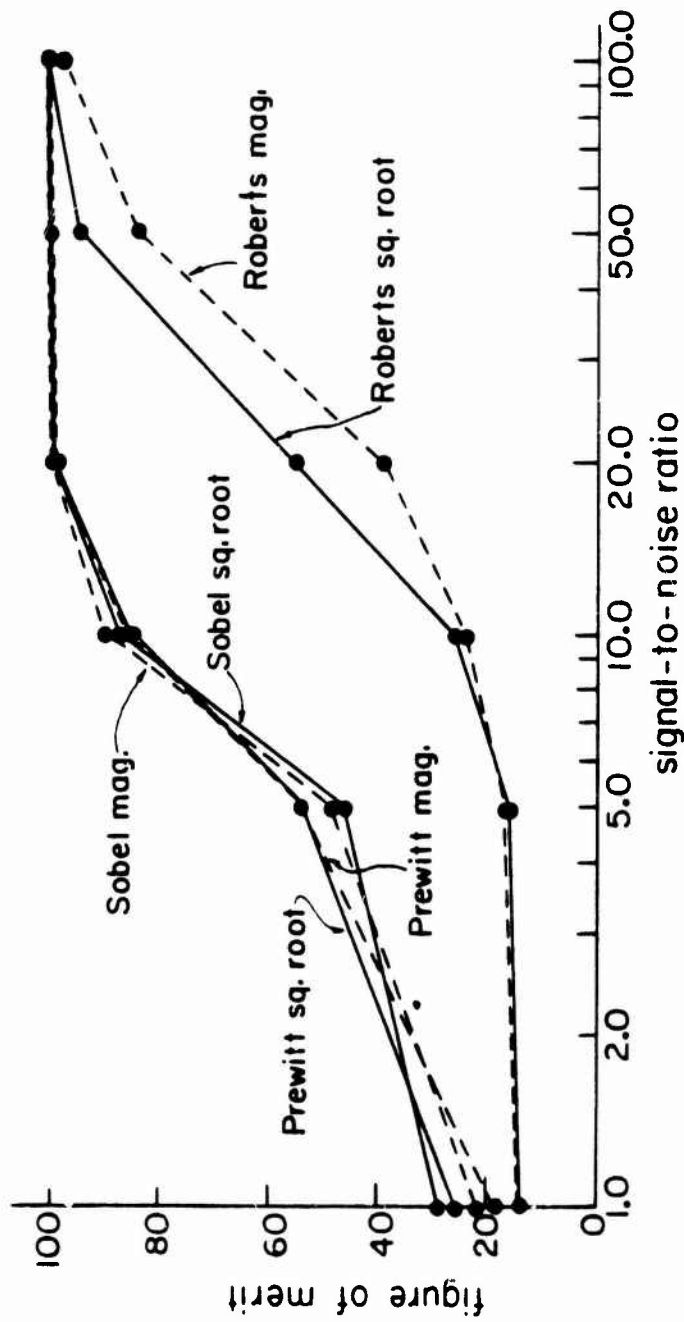
5.2 Experimental Results

The Sobel, Prewitt, compass gradient, Kirsch, 3-level and 5-level operators are evaluated using the figure of merit defined previously. The test images are generated in the form of ideal steps with vertical or diagonal orientations. The height is $h = 25$. Gaussian noise is added to the ideal step with signal-to-noise ratios 1.0, 5.0, 10.0, 20.0, 100.0, respectively. Each edge detector is applied on the different test images, and the threshold t is varied until the figure of merit is maximum. Plots of the figure of merit as a function of signal-to-noise ratio are shown in Figures 5.3 and 5.4. The figures of merit generally follow expected trends: small for low signal-to-noise ratios and large in the opposite case. Some of the edge detection methods are superior to others



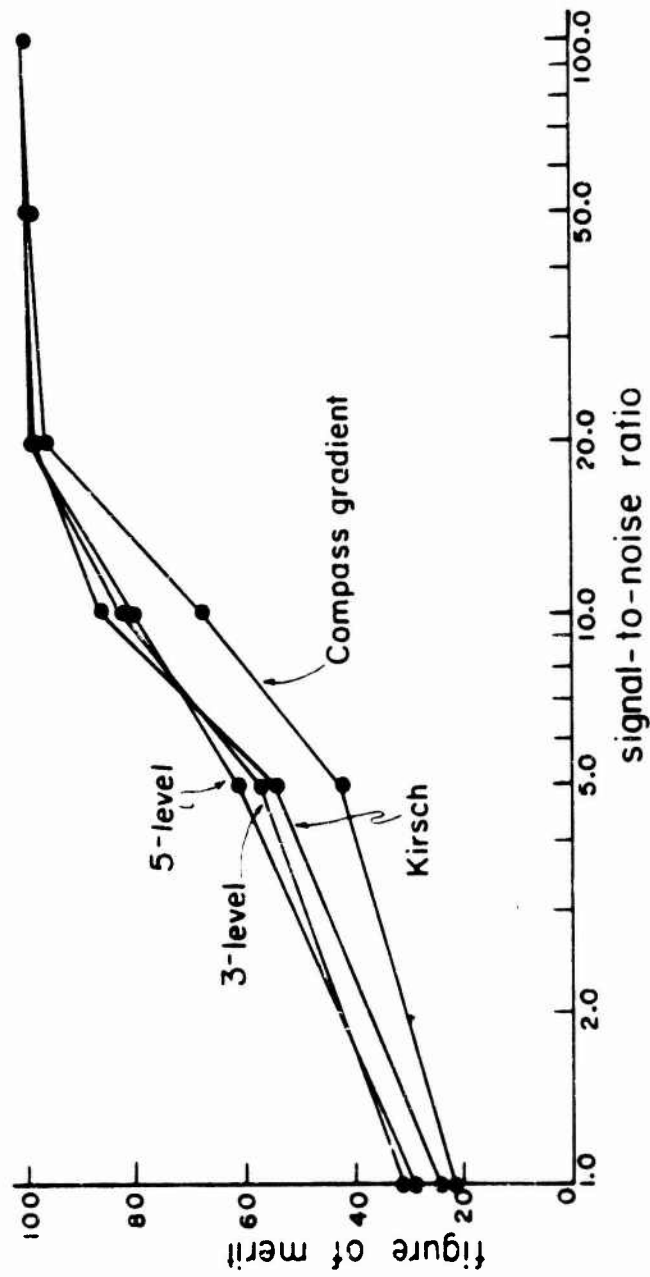
a) vertical edge

Figure 5.3. Figure of a merit as a function of signal-to-noise ratio for simple differential operators



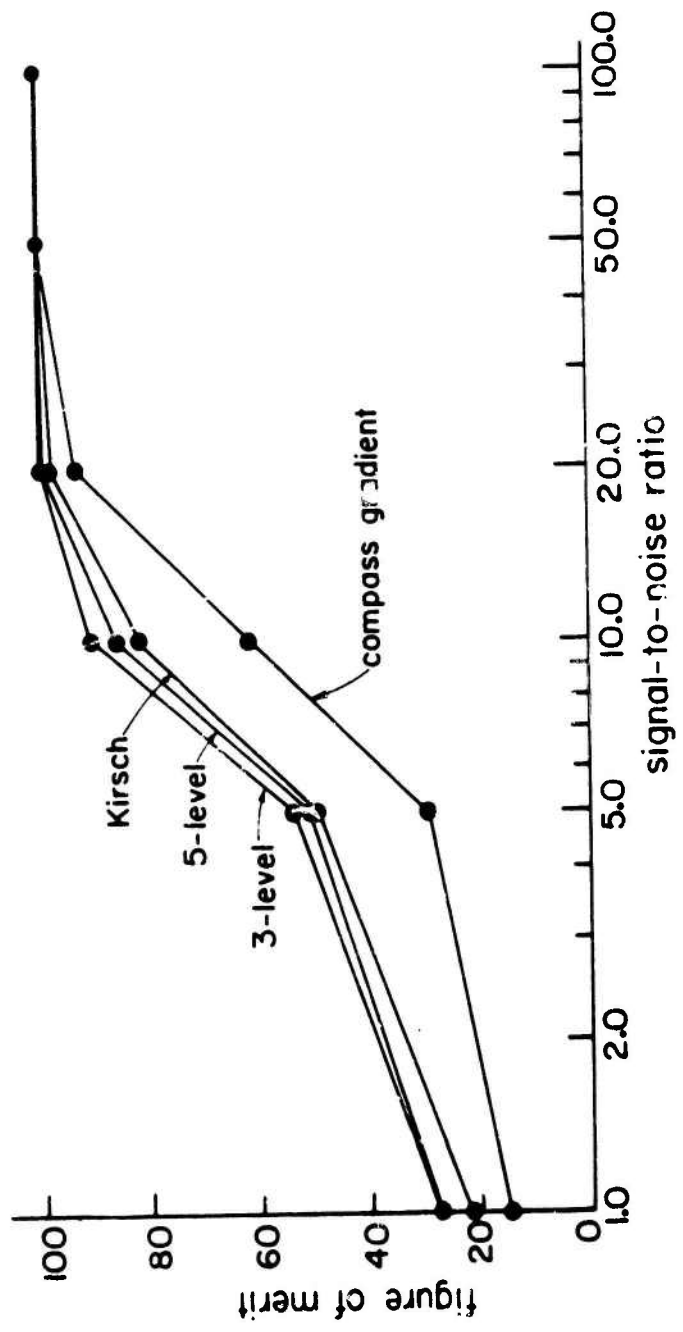
b) diagonal edge

Figure 5.3. (Continued)



a) vertical edge

Figure 5.4. Figure of merit as a function of signal-to-noise ratio for template matching operators



b) diagonal edge

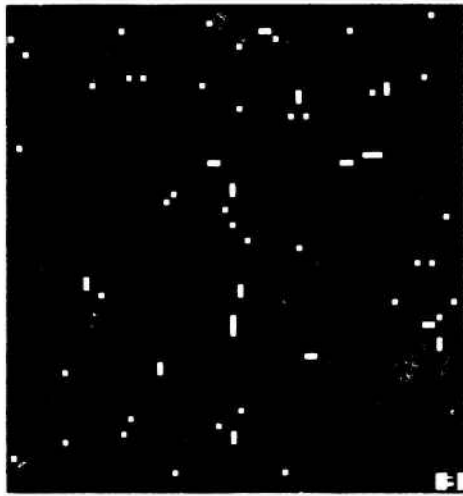
Figure 5.4. (Continued)

for all test images. Examples of the edge maps, obtained in the previous experiments, are shown in Figure 5.5. It should be noticed that the figures of merit are correlated with visual quality of the edge maps.

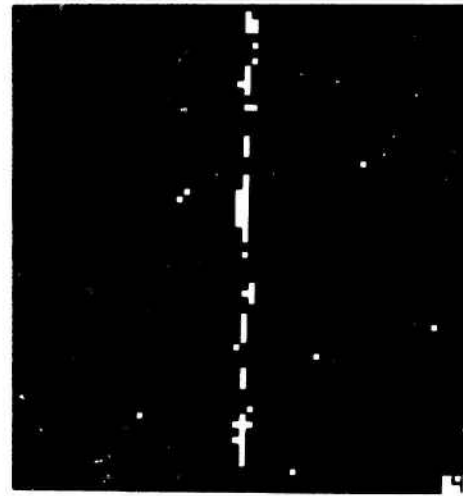
The figures of merit plotted in Figures 5.3 and 5.4 can be related to the response of an edge detector to displaced edges, shown in Figure 2.10, and to the operating characteristics of an edge detector, as shown in Figures 3.2 and 3.3. The figure of merit is large when the edge detectors have good performance in the presence of noise, and when the edge detectors suppress non central edges efficiently.

5.3 Conclusion

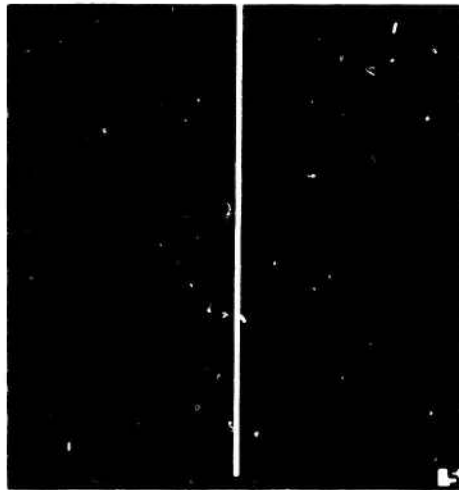
In general, the results obtained in Chapters 3, 4 and 5 show that the 3-level operator has better performance than any of the other edge detectors. Its performance can be compared only to the performance of the Prewitt operator. The advantage of the 3-level operator is that it has almost the same performance for all edge orientations, while the advantage of the Prewitt is that it requires less computation effort, especially if the square root is replaced by the sum of magnitudes.



a) Prewitt square root
vertical edge, SNR=1



b) Prewitt square root
vertical edge, SNR=10

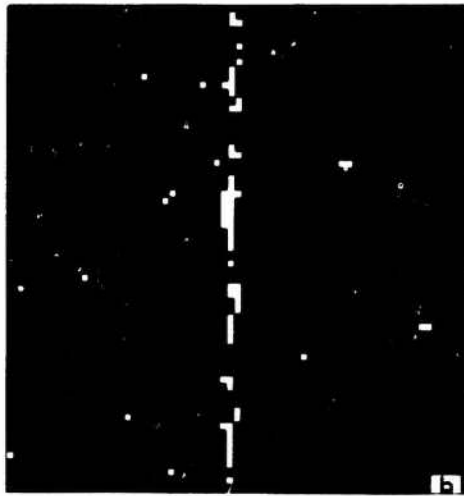


c) Prewitt square root
vertical edge, SNR=100



d) Prewitt square root
diagonal edge, SNR=10

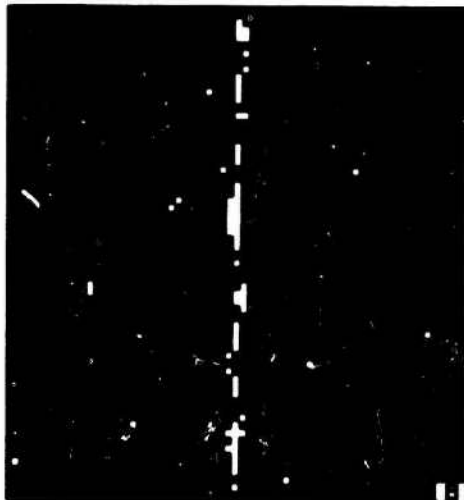
Figure 5.5. Edge maps for 2x2 and 3x3 operators



e) Sobel square root
vertical edge, SNR=10



f) Roberts square root
vertical edge, SNR=10



g) 3-level
vertical edge, SNR=10



h) Kirsch
vertical edge, SNR=10

Figure 5.5. (Continued)

Chapter 6

New Edge Enhancement/Thresholding Methods

The analysis introduced so far has been concerned with the evaluation of existing edge detection operators. This evaluation is one of two objectives of the dissertation. The other objective being to introduce new edge detection techniques and to evaluate their performance. In this chapter, some new trends in edge enhancement/thresholding are given. In Chapter 7, a new edge fitting algorithm is discussed.

There are some modifications that can be introduced to the edge enhancement/thresholding operators, such as changing the mask size, weighting the mask elements, and using an adaptive thresholding procedure. Before introducing these modifications, it is useful to evaluate their effects and to decide if they actually improve the edge detector performance. This will be the subject of the following sections. In Section 6.1, the effect of increasing the mask size is evaluated. In Section 6.2, the effect of weighting the mask elements is discussed. In Section 6.3, some adaptive edge thresholding methods are introduced.

6.1 Effect of Changing Mask Size

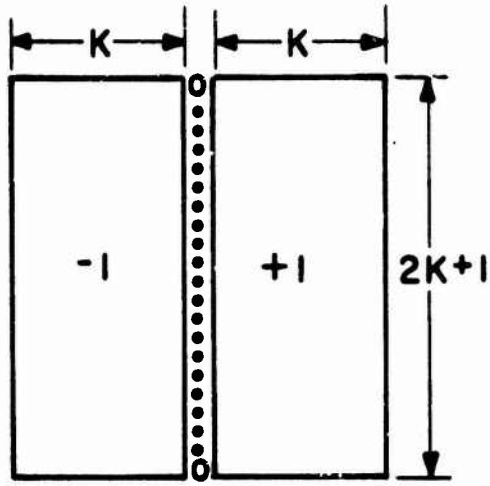
The 3x3 edge detectors can be considered as a special case of general $(2K+1) \times (2K+1)$ edge detectors. Extension of the two masks of the Prewitt operator, is shown in Figures 6.1a and b. Also, the set of four masks of Figure 6.1 represent an extension of the 3-level operator. Increasing the mask size will affect edge detector performance in two ways. First, the operator will be less sensitive to noise because it bases its decision on a larger number of pixels. Second, the edge detector will have a lower resolution. A discussion of these two effects in the case of the 3-level and the Prewitt operators is given in the following paragraphs.

The performance of the $(2K+1) \times (2K+1)$ operators in the presence of noise, can be evaluated using the statistical model of Chapter 3. In the case of the 3-level operator, the covarianc matrix $\underline{\underline{\Sigma}}$, and mean vector $\underline{\underline{\tilde{G}}}_V^*$ are of the form

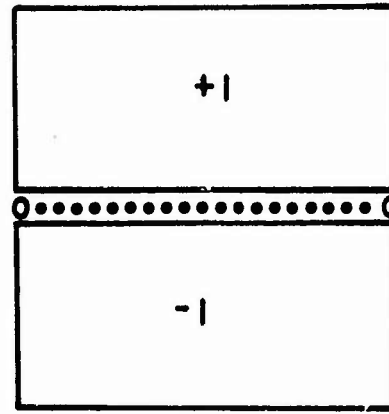
$$\underline{\underline{\Sigma}} \cdot \sigma^2 = \begin{bmatrix} 2K(2K+1) & 2K(K+1) & 0 & -2K(K+1) \\ -2K(K+1) & 2K(2K+1) & 2K(K+1) & 0 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \quad (6.1)$$

$$\underline{\underline{\tilde{G}}}_V^* = h[K(2K+1) \quad k(K+1) \quad 0 \quad -K(K+1)]^T \quad (6.2)$$

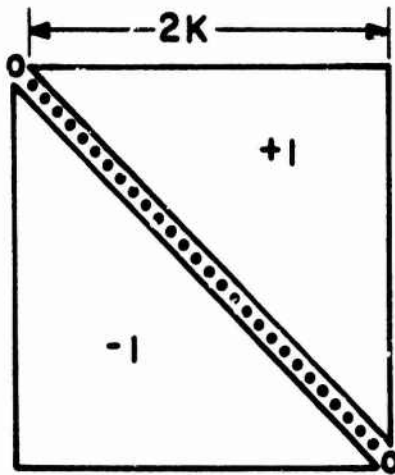
$\underline{\underline{\tilde{G}}}_V^*$ denotes the mean vector for a vertical edge



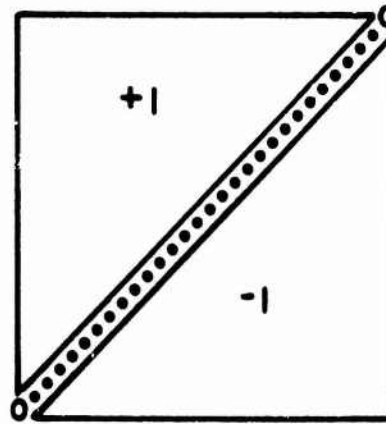
a) vertical



b) horizontal



c) positive diagonal



d) negative diagonal

Figure 6.1. Extended masks for the Prewitt and the 3-level operators

In the case of the Prewitt operator, the output of the vertical and horizontal masks are independent Gaussian random variables, with covariance matrix $\underline{\Sigma}$, and the mean vector $\underline{\tilde{G}}_v$, in the form

$$\underline{\Sigma} = \begin{bmatrix} 2K(2K+1) & 0 \\ 0 & 2K(2K+1) \end{bmatrix} \quad (6.3)$$

$$\underline{\tilde{G}}_v = h[K(2K+1) \quad 0]^T \quad (6.4)$$

The probabilities of detection and false detection can be evaluated as in Chapter 3. Plots of the edge detector operating characteristics for a signal-to-noise ratio of 1.0, and operator mask sizes of 5x5, 7x7, and 9x9 are given in Figure 6.2. From these plots, it is clear that the performance of the 3-level operator is better than the performance of the Prewitt operator for diagonal edges, while it is slightly less than the performance of the Prewitt operator for vertical edges. Also, it can be easily noticed that performance improves as the mask size increases. On the other hand, increasing the mask size will reduce the edge detector resolution. This effect can be shown by plotting edge detector output as a function of the distance between the edge and the center of the operator. Plots of the normalized outputs of 3x3, 5x5,

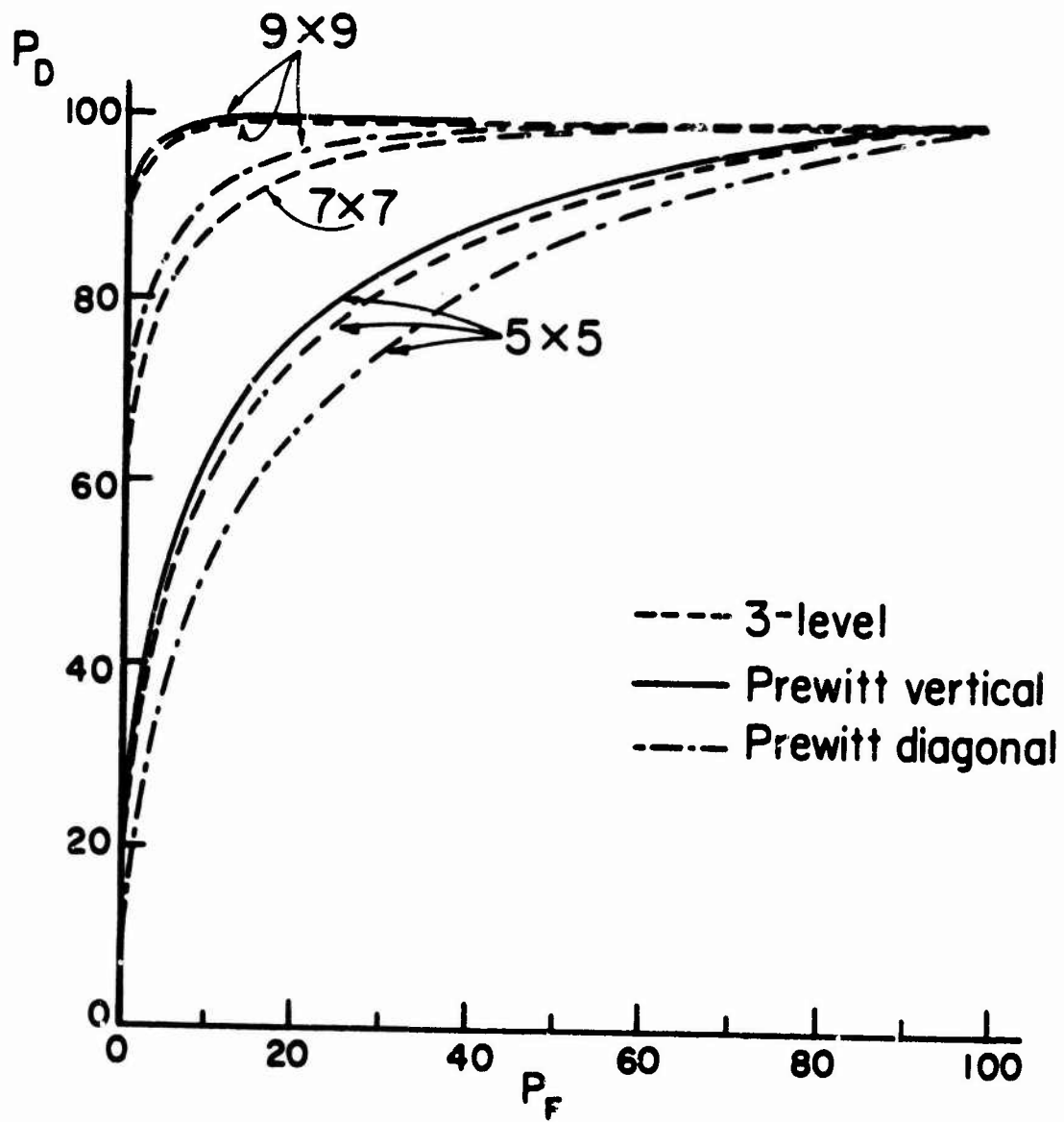


Figure 6.2. Probability of detection versus probability of false detection for extended Prewitt and 3-level operators

7x7, and 9x9 mask operators, in the case of a vertical edge, are shown in Figure 6.3. It is clear that, as the mask size increases, the region over which the edge is detected increases. This will reduce the operator's ability to detect the finer details of the image.

The previous two effects can be measured simultaneously by using the figure of merit defined in Chapter 5. The 3-level and the Prewitt operators are applied on the test images containing a vertical and a diagonal edge. The figure of merit is plotted as a function of the signal-to-noise ratio. These curves are shown in Figure 6.4. The results agree with the previous analysis: for low signal-to-noise ratio, the operators with large mask size have better performance because they are less sensitive to noise, which is a dominant factor in this case, while for large signal-to-noise ratio, the operators with small mask size have better performance because they are more accurate in detecting edge location. Examples of the edge maps for the vertical edge with $SNR = 1.0$ are shown in Figure 6.5. These examples give a visual indication of the improvement achieved by increasing the mask size.

Since the 3-level and the Prewitt operators achieve an almost optimum performance while using simple computation procedures, the performance of these operators can be used

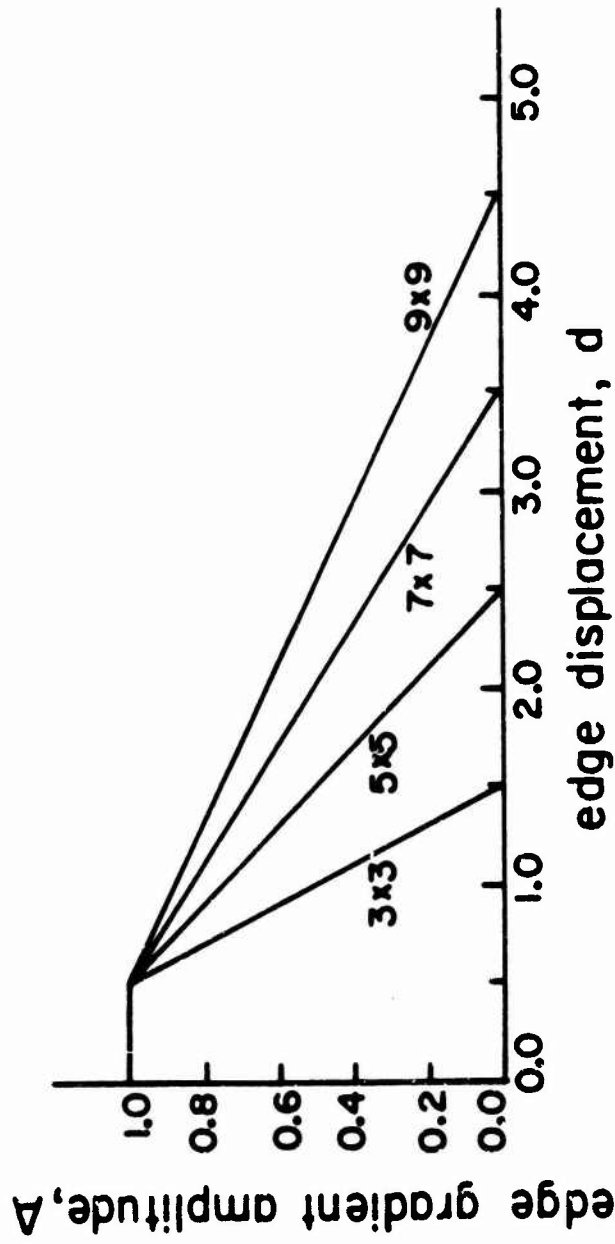
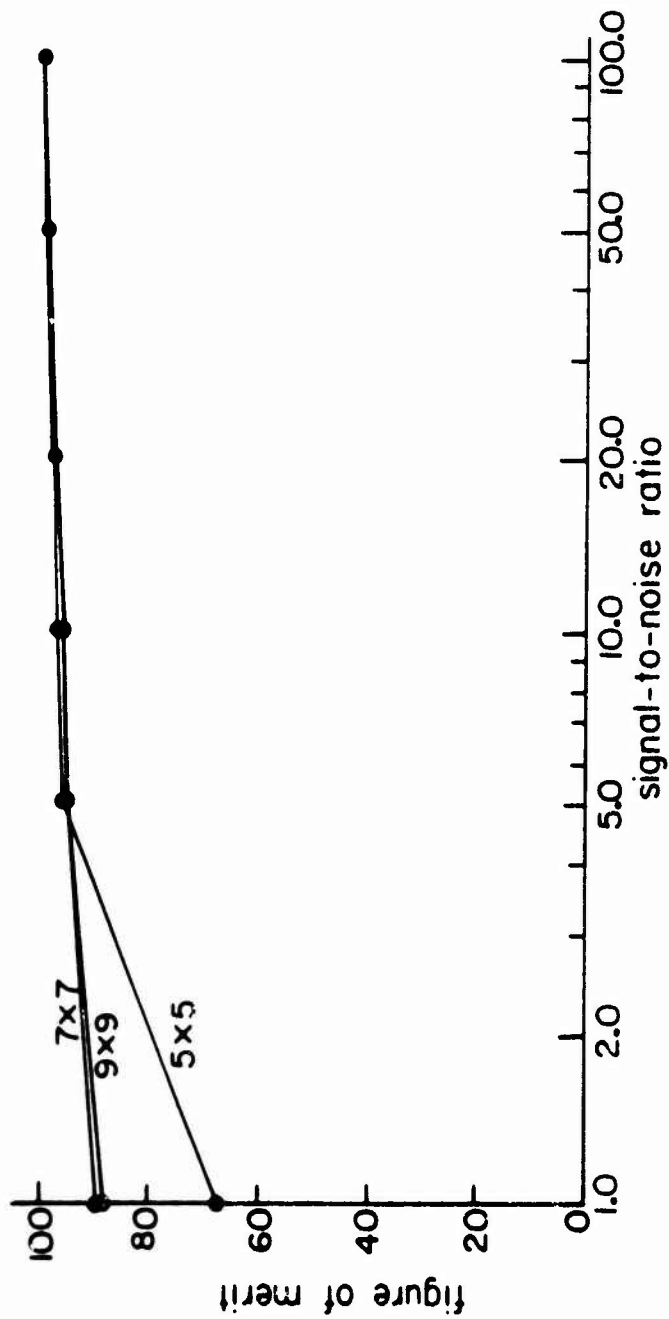
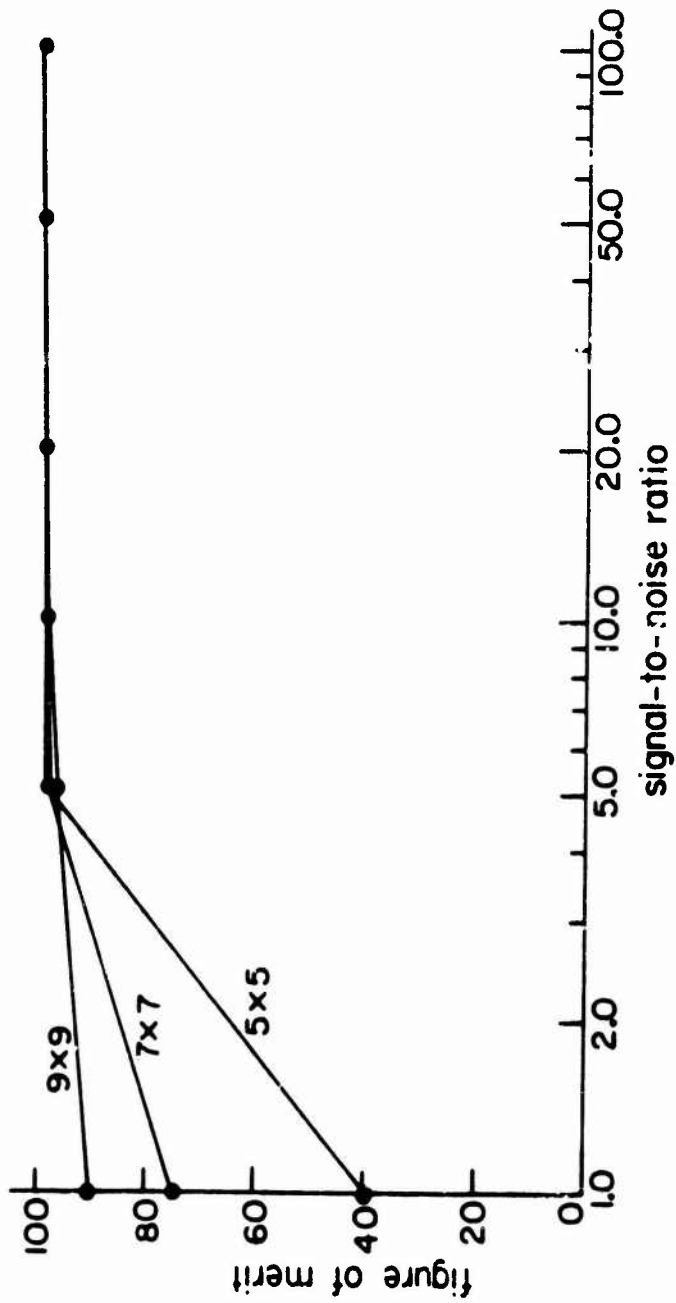


Figure 6.3. Edge gradient amplitude response as a function of edge displacement for extended Prewitt and 3-level operators



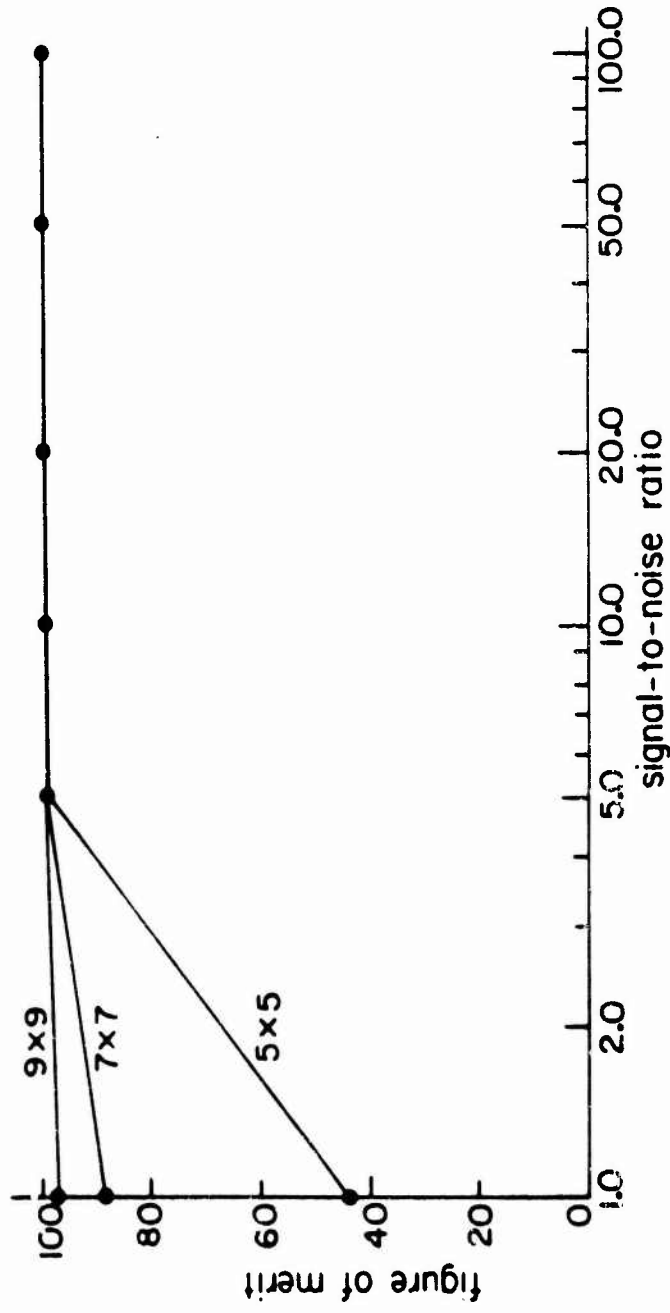
a) vertical edge, Prewitt and 3-level

Figure 6.4. Figure of merit as a function of signal-to-noise ratio for extended Prewitt and 3-level operators



b) diagonal edge, Prewitt

Figure 6.4. (Continued)



c) diagonal edge, 3-level

Figure 6.4. (Continued)



a) 5x5 mask



b) 7x7 mask



c) 9x5 mask

Figure 6.5. Edge maps for extended Prewitt operator, vertical test image with SNR=1

as a standard to which any other edge detector performance should be compared. As an example, the performance of the conventional 69 pixel Hueckel operator is compared with the performances of a 7x7 or a 9x9 mask operators in Appendix A. This comparison indicates that the 3-level and the Prewitt operators has better performances than the Hueckel operator.

6.2 Use of Weighted Masks

The resolution of edge detectors with large mask size can be improved by weighting the mask elements, such that they are maximum near the mask center and decrease to zero as they approach the mask periphery. There are many examples of weighted masks that can be used in edge detection. Argyle [23] has proposed a split Gaussian function defined in one dimension as

$$h(x) = \begin{cases} \frac{1}{\sqrt{2\pi}k} \exp\left(-\frac{x^2}{2k^2}\right) & x > 0 \\ 0 & x = 0 \\ \frac{-1}{\sqrt{2\pi}k} \exp\left(-\frac{x^2}{2k^2}\right) & x < 0 \end{cases} \quad (6.5)$$

where k is a spread constant. Macleod [24] introduced a continuous Gaussian function; a special case of the Macleod function is given by

$$H(x,y) = \exp\left(-\frac{x^2}{t^2}\right) \left\{ \exp\left[-\left(\frac{x-p}{F}\right)^2\right] - \exp\left[-\left(\frac{x+p}{P}\right)^2\right] \right\} \quad (6.6)$$

where p and t are spread constants. Another example of the

weighting functions is the polynomial

$$H(x,y) = \begin{cases} \frac{1}{1+\alpha y^2} & \frac{1}{1+\alpha x^2} \dots \dots \dots x > 0 \\ 0 & \dots \dots \dots x = 0 \\ -\frac{1}{1+\alpha y^2} & \frac{1}{1+\alpha x^2} \dots \dots \dots x < 0 \end{cases} \quad (6.7)$$

where α is an adjustable scaling factor. The elements of the previous weighted masks are not integers, and thus require more computation time compared with the 3-level simple mask. This problem can be avoided if the weighted mask is chosen to be the pyramid shaped mask shown in Figure 6.6.

To test the resolution of the different weighted masks, the outputs of 7x7 weighted mask operators for displaced vertical edges are plotted in Figure 6.7. In this experiment, $k = p = t = 4.0^*$ and $\alpha = 1/9$. The results show that the pyramid-shaped mask has the best resolution followed by the polynomial, the Argyle, the simple 3-level, and finally the Macleod weighted mask.

The statistical model of Chapter 3 can be used to evaluate the performance of the weighted mask operators. As an example, for the weighted Prewitt operator, the performance will depend on the ratio between the ideal edge output (a), and the noise standard deviation (σ_r). The

*These are the parameters suggested by Fram and Deutsch in their paper [13].

-1	-1	-1	-1	-1	-1	0	1	1	1	1	1	1
-1	1
-1	.	-k	-k	-k	-k	0	k	k	k	k	.	1
-1	.	-k	.	.	.	0	.	.	.	k	.	1
-1	.	-k	.	-K+1	-K+1	0	K-1	K-1	.	k	.	1
-1	.	-k	.	-K+1	-K	0	K	K-1	.	k	.	1
-1	.	-k	.	-K+1	-K	0	K	K-1	.	k	.	1
-1	.	-k	.	-K+1	-K	0	K	K-1	.	k	.	1
-1	.	-k	.	-K+1	-K+1	0	K-1	K-1	.	k	.	1
-1	.	-k	.	.	.	0	.	.	.	k	.	1
-1	.	-k	-k	-k	-k	0	k	k	k	k	.	1
-1	1
-1	-1	-1	-1	-1	-1	0	1	1	1	1	1	1

Figure 6.6. Pyramid operator

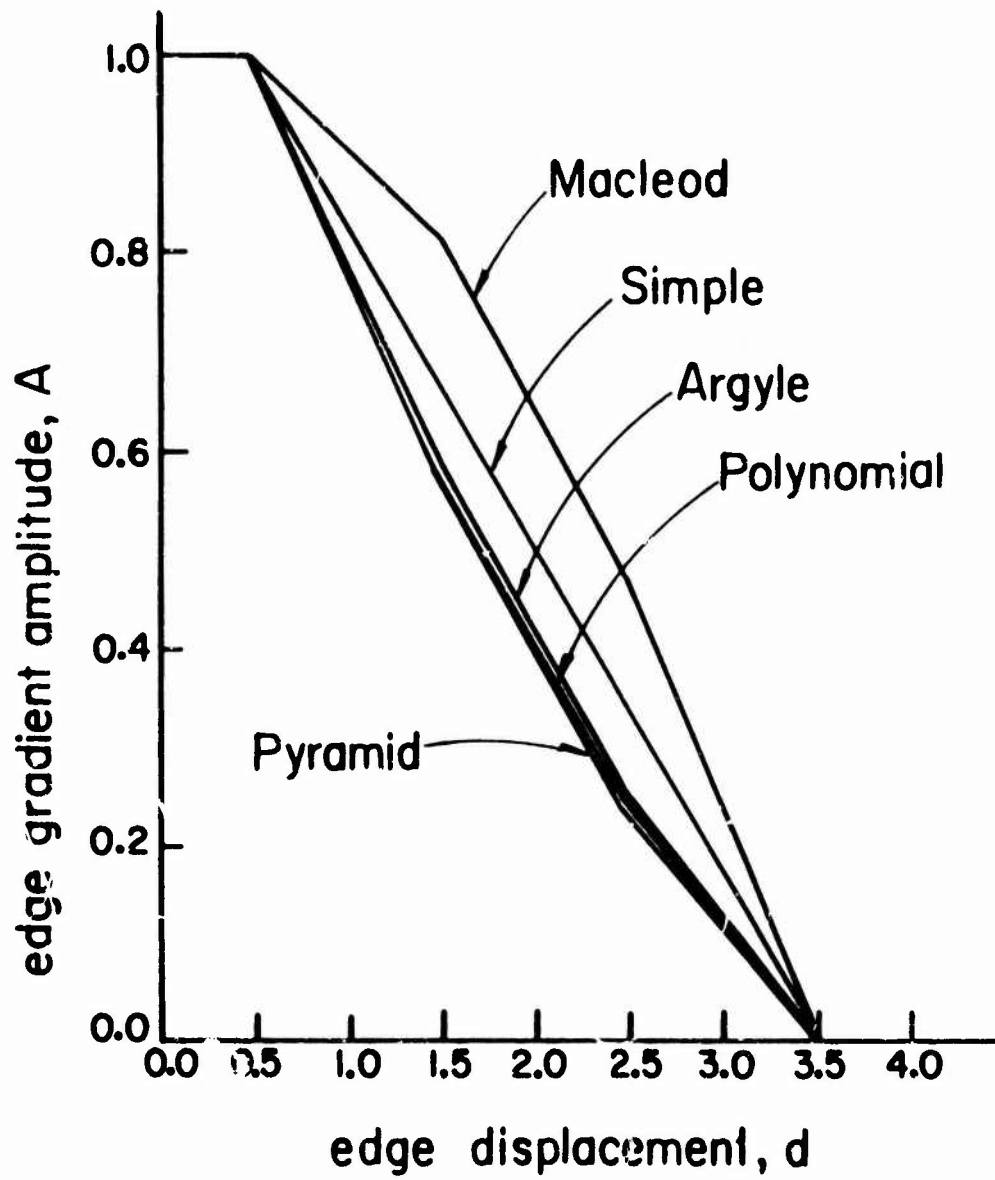


Figure 6.7. Edge gradient amplitude response as a function of edge displacement for weighted 7x7 operators

larger this ratio, the better the performance. In Table 6.1, the values of a/σ_r for the 7x7 weighted mask operators are given. These ratios, and hence the performance of the weighted mask operators depend on the shape of the weighting function used. In general, the edge detector will have a better performance in the presence of noise if the mask elements are more uniform, with the optimum performance achieved by using equal mask elements.

The different weighted-mask edge detectors can be evaluated using the figure of merit of Chapter 5. In this experiment, the vertical edge test image is used to evaluate the Argyle, Macleod, polynomial and pyramid shaped operators with a mask size 7x7. Results are shown in Figure 6.8. It is clear that, excluding the Macleod operator, most of the weighted mask operators have approximately identical performances. The inferior performance of the Macleod operator can be improved by changing its parameters.

6.3 Use of Adaptive Thresholding

In the previous experiments, the value of the threshold t was found to be a function of the absolute signal levels and the signal-to-noise ratio. In simple test images, t can be a constant for all the image subregions. In real world images, however, a constant threshold should not be used because it will enhance the

Table 6.1.

The Ratio ($\frac{a}{\sigma_I}$) for Weighted Masks

weighting function	3-level simple	Argyle	Polynomial	Macleod	Pyramid
$\frac{a}{\sigma_I}$	21.00	19.39	18.74	18.07	17.52

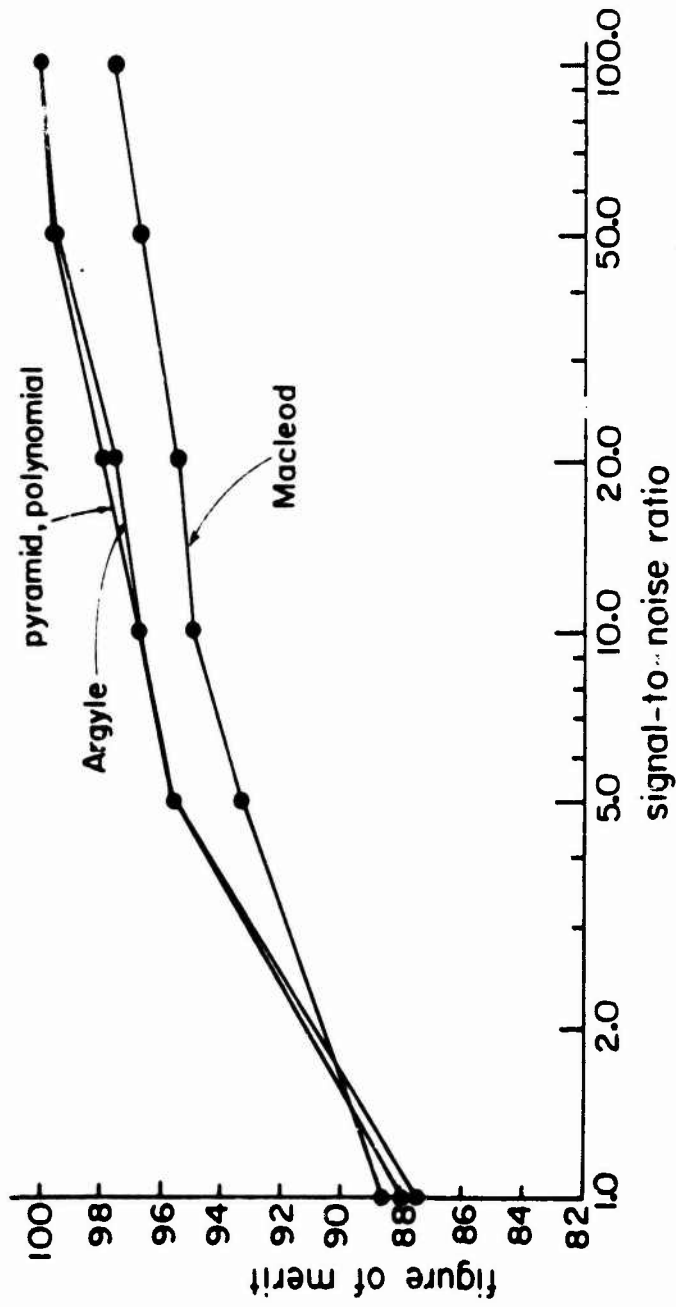


Figure 6.8. Figure of merit as a function of signal-to-noise ratio for weighted 7x7 operators

boundaries between high intensity regions more than the boundaries between low intensity regions. This problem can be avoided if the output of the edge detectors is compared with a function of the subregion intensities. This can be considered as a local adaptive thresholding procedure [7]. Examples of the functions that can be used are the average

$$t = \alpha_1 \sum_{j=1}^J f_j \quad (6.8)$$

the root mean square

$$t = \alpha_2 \left(\sum_{j=1}^J f_j^2 \right)^{\frac{1}{2}} \quad (6.9)$$

and in general

$$t = \left[\alpha_1 \left(\sum_{j=1}^J f_j \right)^2 + \alpha_2 \left(\sum_{j=1}^J f_j^2 \right) \right] \quad (6.10)$$

In Eqs. 6.8 to 6.10., f_1, f_2, \dots, f_J are the pixels intensities, and α_1, α_2 are constants that can be adjusted.

A quantitative evaluation of these adaptive thresholding methods is not simple because it requires the knowledge of the image model. A discussion of the problem will be given in Chapter 8. Some of the experimental results obtained with the adaptive thresholding edge detectors will be shown in Appendix E.

6.4 Conclusion

In this chapter, various modifications in the edge enhancement/thresholding operators have been considered. The purpose of these changes is to achieve a compromise between better resolution and acceptable performance in the presence of noise. It is believed that this compromise should be one of the basic objectives in edge detector design. Other methods that achieve better edge resolution through edge thinning can be found in the works of Rosenfeld [5,25], and Herskovits [26].

Chapter 7

A New Edge Fitting Algorithm

Minimum-error surface fitting techniques have been considered by many as an optimum solution to the edge detection problem. Although this is true theoretically, in practical applications, the surface fitting algorithms suffer from two drawbacks. The first is that the image is usually defined over a sampled domain while most of the surface fitting algorithms are derived for continuous functions. The second is that even assuming the image to be continuous, the optimization procedures require the solution of implicit functions of the edge parameters. This solution can be achieved through iterative procedures, which are time consuming and thus cannot be practically used in edge detection. Usually some approximations are made to avoid this iterative solution. As an example, in the Hueckel operator, the optimization procedure is simplified by using truncated Fourier expansions of the image subregion and the ideal edge model. The effect of this approximation on the optimality of the solution cannot be easily evaluated.

The previous difficulties can be avoided by using edge fitting algorithms based on the discrete image model. One of these algorithms will be introduced in the following sections. In Section 7.1, a one-dimensional edge fitting algorithm is discussed. In Section 7.2, the model is extended to the more important case of two-dimensional edge fitting. In Section 7.3, evaluation of the edge fitting algorithm performance is given.

7.1 One-Dimensional Edge Fitting

The problem of one-dimensional edge fitting can be stated as follows: given a continuous function $f(x)$ defined for $-b \leq x \leq b$, it is required to find a piecewise linear function $s_p(x)$ such that the error

$$E_p = \int_{-b}^b (s_p(x) - f(x))^2 dx \quad (7.1)$$

is minimum. The problem can be simplified by assuming that the function $s_p(x)$ is centered around the origin, as shown in Figure 7.1. In this case $s_p(x)$ is given by

$$s_p(x) = \begin{cases} a - \Delta x_0 & -b \leq x < -x_0 \\ a + \Delta x & -x_0 \leq x < x_0 \\ a + \Delta x_0 & x_0 \leq x \leq b \end{cases} \quad (7.2)$$

where a is the average value of $s_p(x)$, Δ is the ramp slope, and x_0 is the half ramp width. These three parameters are

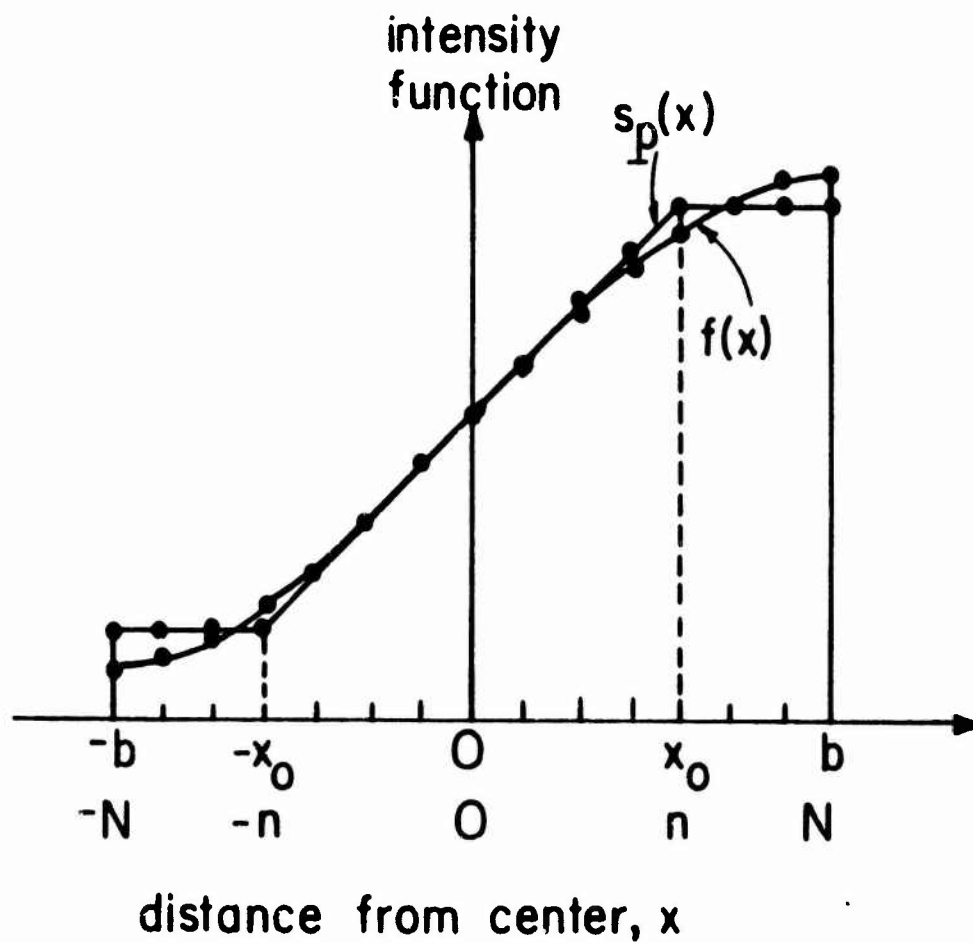


Figure 7.1. One-dimensional edge model

combined in the vector

$$\underline{p} = [a \quad \Delta \quad x_0]^T \quad (7.3)$$

The value of \underline{p} that minimizes Eq. 7.1 is obtained by solving the set of equations

$$\frac{\partial E}{\partial a} = 0 \quad (7.4a)$$

$$\frac{\partial E}{\partial \Delta} = 0 \quad (7.4b)$$

$$\frac{\partial E}{\partial x_0} = 0 \quad (7.4c)$$

Substituting in the previous equations, the optimum parameter vector \underline{p} is given by

$$a = \frac{1}{2b} \int_{-b}^b f(x) dx \quad (7.5a)$$

$$\int_{x_0}^{x_0} x f(x) dx = 3\Delta x_0^2 b + \frac{1}{3}\Delta x_0^3 \quad (7.5b)$$

$$\int_{-b}^{-x_0} f(x) dx - \int_{x_0}^b f(x) dx = \Delta x_0 (b - x_0) \quad (7.5c)$$

It is clear that even for this simplified case, the solution is based on implicit functions of x_0 and Δ . Instead of solving Eqs. 7.5b and c through an iterative procedure, it has been found that reformulating the problem in the discrete domain will save computation time, while giving a solution that is feasible.

In the discrete domain, the functions $f(x)$ and $s_p(x)$ are defined only for the set of points $\{-N, \dots, 0, \dots, N\}$. In all of the following discussions, the ramped part of $s_p(x)$ is assumed to start and end at sample points $-n$ and n respectively. This assumption simplifies the computation without a substantial change in the accuracy of the results. The curve fitting procedure reduces to finding the parameter vector

$$\underline{p} = [a \quad \Delta \quad n]^T \quad (7.6)$$

such that the error

$$E_p = \sum_{i=-N}^N (s_p(i) - f(i))^2 \quad (7.7)$$

is minimum. Since n assumes a finite number of integer values, the minimization problem can be solved by repeating the computation for each value of n and choosing the value of n that minimizes E . In addition, by differentiating with respect to a , it can be shown that for any value of n , the optimum a is independent of n and is given by the average:

$$a = \frac{1}{2N+1} \sum_{i=-N}^N f(i) \quad (7.8)$$

Substituting the values of $s_p(i)$ in Eq. 7.7 and arranging the terms, E_p can be expressed in the form

$$E_p = C_0 + C_1 \Delta + C_2 \Delta^2 \quad (7.9)$$

where

$$C_0 = \sum_{-N}^N (a-f(i))^2 \quad (7.10a)$$

$$C_1 = 2n \sum_{-N}^{-(n+1)} f(i) - 2 \sum_{-n}^n i f(i) - 2n \sum_{n+1}^N f(i) \quad (7.10b)$$

and

$$C_2 = \sum_{-N}^{-(n+1)} n^2 + \sum_{-n}^n i^2 + \sum_{n+1}^N n^2 \quad (7.10c)$$

Equation 7.9 can be minimized by choosing

$$\Delta = -\frac{C_1}{2C_2} \quad (7.11)$$

and for this value of Δ , E_p is given by

$$E_p = C_0 - \frac{C_1^2}{4C_2} \quad (7.12)$$

One-dimensional edge fitting can be achieved by the following procedure: given a function $f(i)$ defined over the range $[-N, N]$, the average (a) is computed using Eq. 7.8. Assuming that $f(i)$ can be fitted to a ramp $s_p(i)$ with width n , the optimum value of Δ and the corresponding minimum error E_p are computed using Eqs. 7.11 and 7.12. The computation is repeated for different values of n , and the minimum error in each case is compared. The values of n and Δ that result in a global minimum error are chosen as the edge parameters. Finally, the acceptance of the edge

fitting can be determined from the signal-to-noise ratio, Δ^2/E_{\min} . If this ratio is larger than a threshold t , the edge fitting is accepted.

7.2 Two-Dimensional Edge Fitting

The previous analysis can be extended to two-dimensional edge fitting. In this case, the image function $f(i,j)$ defined over a subregion is compared with an ideal edge model $S_p(i,j)$, where

$$\underline{p} = [a \quad \theta_i \quad \Delta \quad n]^T \quad (7.13)$$

is the parameter vector. The variables a , θ_i , Δ and n are defined as in Section 7.1 where θ_i indicates the edge orientation. In the following experiments, θ_i assumes one of four basic orientations, horizontal, vertical and the two diagonals. The effect of this approximation on the accuracy of the edge fitting, will be discussed in Section 7.3. The edge fitting is achieved by changing the edge parameter vector \underline{p} to minimize the error

$$E_p = \sum_{i=-N}^N \sum_{j=-N}^N (S_p(i,j) - f(i,j))^2 \quad (7.14)$$

Following the analysis in Section 7.1, it can be shown that, for the minimum error, the parameter a is given by

$$a = \frac{1}{(2N+1)^2} \sum_i \sum_j f(i,j) \quad (7.15)$$

The parameters θ_i and n can be changed in finite steps, and for each combination of θ_i and n , the error E_p is in the form

$$E_p = C_0 + C_1\Delta + C_2\Delta^2 \quad (7.16)$$

where

$$C_0 = \sum_i \sum_j (a-f(i,j))^2 \quad (7.17)$$

and

$$C_1 = 2n \sum_{i=-N}^{-(n+1)} F(i) - 2 \sum_{i=-n}^n iF(i) - 2n \sum_{i=n+1}^N F(i) \quad (7.18a)$$

$$C_2 = (2N+1) \left[2(N-n)n^2 + \sum_{i=-n}^n i^2 \right] \quad (7.18b)$$

for vertical and horizontal ramps, while

$$C_1 = 2n \sum_{i=-N}^{-(n+1)} [F(i)+F(i+\frac{1}{2})] - \sum_{i=-n}^{-1} [iF(i)+(i+\frac{1}{2})F(i+\frac{1}{2})] \\ - \sum_{i=1}^n [iF(i)+(i-\frac{1}{2})F(i-\frac{1}{2})] - n \sum_{i=n+1}^N [F(i)+F(i-\frac{1}{2})] \quad (7.19a)$$

$$C_2 = 2(N-n) [2(N-n)+1]n^2 + 2 \sum_{i=1}^n \left\{ [2(N-i)+1]i^2 + [2(N-i)+2](i-\frac{1}{2})^2 \right\} \quad (7.19b)$$

In Eqs. 7.18 and 7.19, the axis is taken perpendicular to the edge side, and $F(i)$ indicates the sum of all the elements at distance i from the edge. Sketches of the masks used for vertical and diagonal edges are shown in

Figure 7.2. Since the expression of E_p is the same for both one- and two-dimensional edge fitting, the values of E_{\min} are still given by Eqs. 7.11 and 7.12. Thus, two-dimensional edge fitting can be achieved by the same procedure described in the previous section. The only changes are that the computation has to be repeated for the different θ_i , and that the values of C_0 , C_1 and C_2 are now given by Eqs. 7.17 to 7.19.

The number of computations required for a 7x7 edge fitting algorithm is 273 additions and 112 multiplications. This can be compared to 152 additions and 1 multiplication needed for a 7x7 template matching operator. The effort needed for accessing the image intensities and comparing the masks' outputs is the same for both operators. The CPU times needed by a PDP-10 KL processor to process a 64x64 image, using the 7x7 edge fitting algorithm and template matching operator, are 18 and 15 seconds respectively.

7.3 Performance Evaluation

The performance of the edge fitting algorithm has been evaluated using three different approaches. First, the output of the edge fitting operators for edges with different orientations and distances from the center are compared. Second, a preliminary evaluation of the performance for noisy edges are given. Third, the figure of merit for the edge fitting algorithm is calculated.

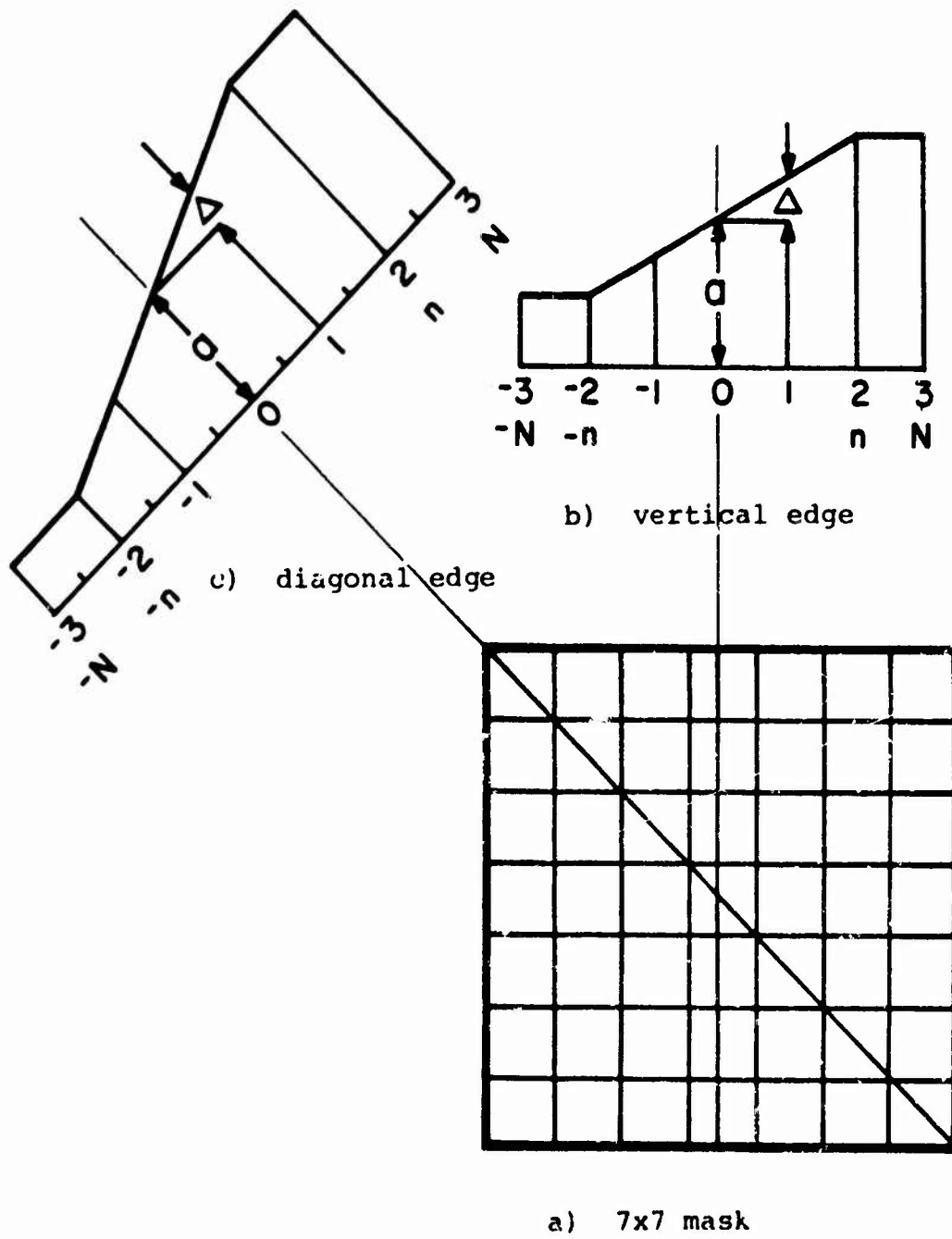


Figure 7.2. Two-dimensional edge models

In the first approach, edge fitting algorithms with mask sizes 5x5, 7x5 and 9x9 are used to process image subregions containing ideal central edges with variable orientation and ideal vertical edges with varying distance from the mask center. Plots of $\sqrt{E_{\min}}/\Delta$ for the previous two cases are shown in Figures 7.3 and 7.4 respectively. In these curves, the abrupt jumps in $\sqrt{E_{\min}}/\Delta$ occur when Δ changes suddenly. This occurs when the width (n) of the edge model that fits the image data is changed. From Figure 7.3, it is obvious that the edge fitting algorithm is not isotropic; the algorithm has the best performance for a vertical edges, it is less sensitive to edges with orientation $\pi/9 \leq \phi \leq \pi/6$, the performance begins to improve again as ϕ approaches $\pi/4$. Also, it should be noticed that the output for $\pi/4$ is not zero. This is because the edge model used does not include a diagonal step which corresponds to ramp width $n = 1/2$. The diagonal edges with fractional ramp width were excluded to save computation effort, and to keep the numbers of edge prototypes equal for both the vertical and the diagonal edge models. The curves in Figure 7.4 show that the error $\sqrt{E_{\min}}/\Delta$ increases sharply as the edge is displaced off center. This feature prevents the multiple detection of the same edge point. The threshold of the edge fitting algorithm can be chosen to allow the detection of central edges with a specified minimum edge height, while

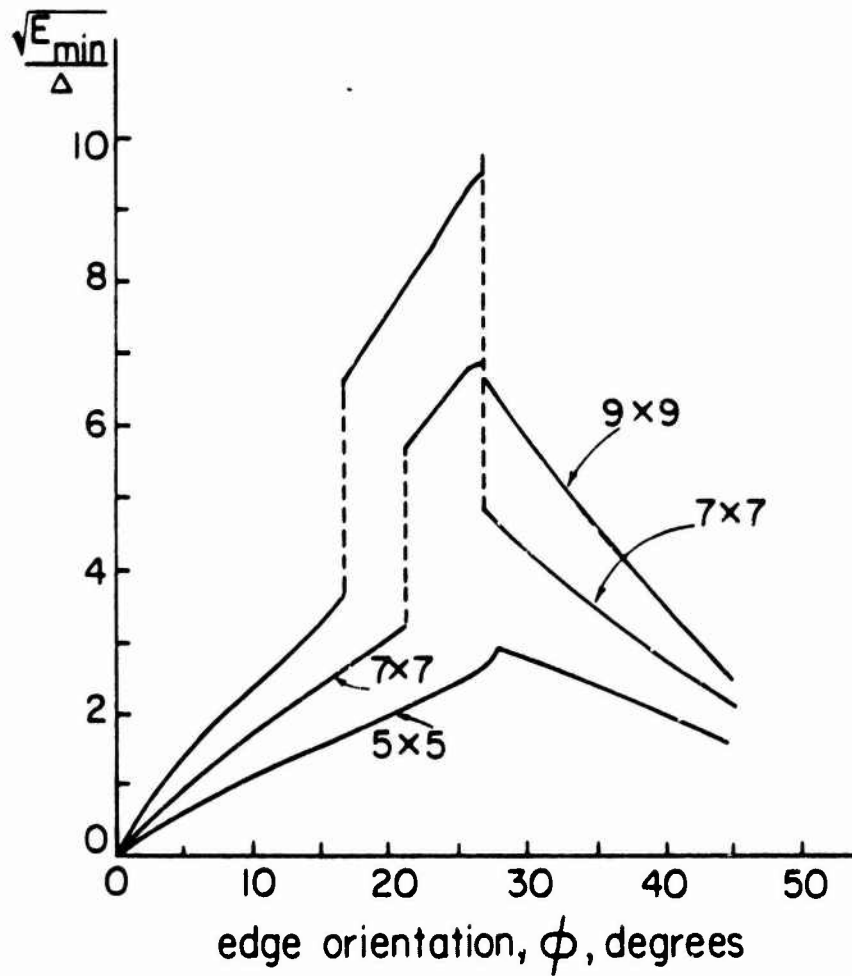


Figure 7.3. Edge fitting normalized error $\sqrt{E_{\min}}/\Delta$, as a function of actual edge orientation

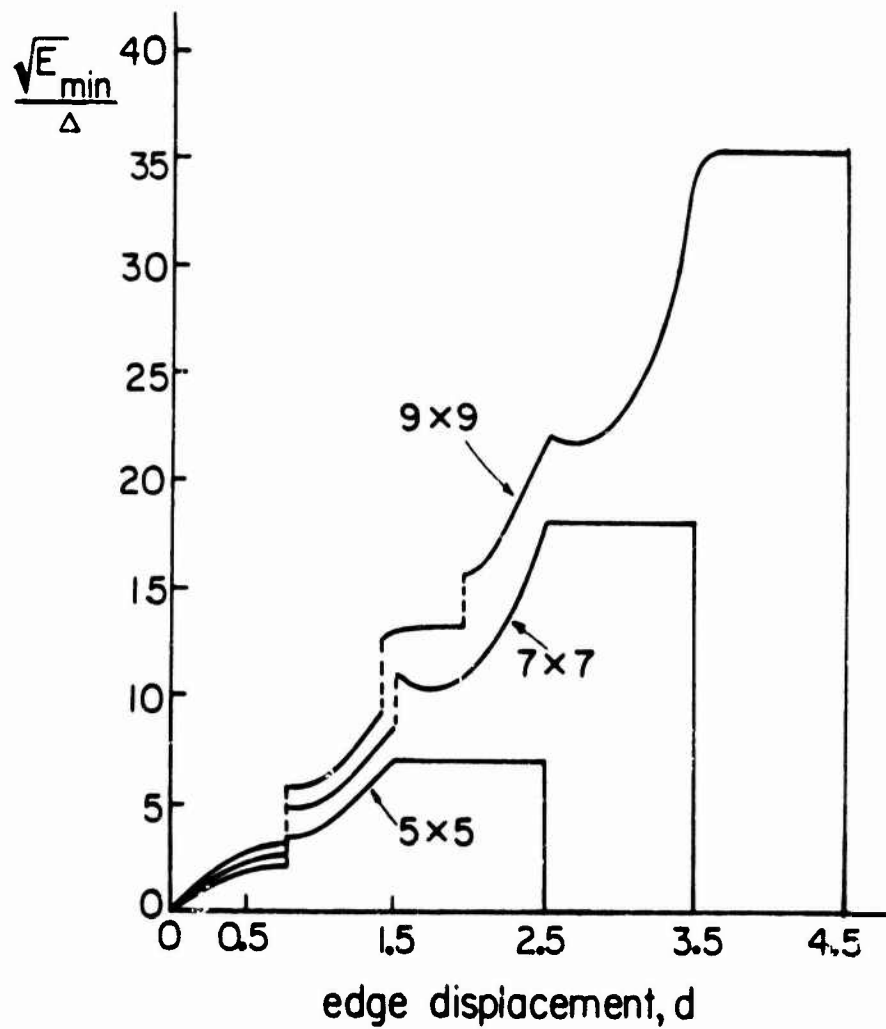


Figure 7.4. Edge fitting normalized error $\frac{\sqrt{E_{\min}}}{\Delta}$, as a function of edge displacement

suppressing displaced edges. Also, it should be noticed that by increasing the number of discrete angles (θ_i), the edge fitting performance will become more uniform. It seems, however, that this change is not necessary, because the performance of the edge fitting algorithm with four basic orientations is sufficiently accurate for all practical applications.

The statistical analysis introduced in Chapter 3 can be used to evaluate the edge fitting algorithm. Derivations of the probability density functions of the coefficients C_0 , C_1 and C_2 and of the error E_p , are straightforward. These derivations are not needed, however, because as a result of the large mask sizes used in the edge fitting algorithms, the noise is usually averaged out. The decision strategy can be derived from the deterministic analysis given previously. To prove the validity of this assumption, the values of $\sqrt{E_{\min}}/\Delta$ are plotted as a function of the edge orientation in the case of a noisy central edge. The results are shown in Figure 7.5. The edge fitting mask is 7x7 and the signal-to-noise ratios are 1.0, 10.0 and 100.0. It should be noticed that for practical levels of SNR, the effect of noise is negligible.

The edge fitting algorithms, with mask sizes 5x5, 7x7 and 9x9, have been evaluated using the figure of merit of

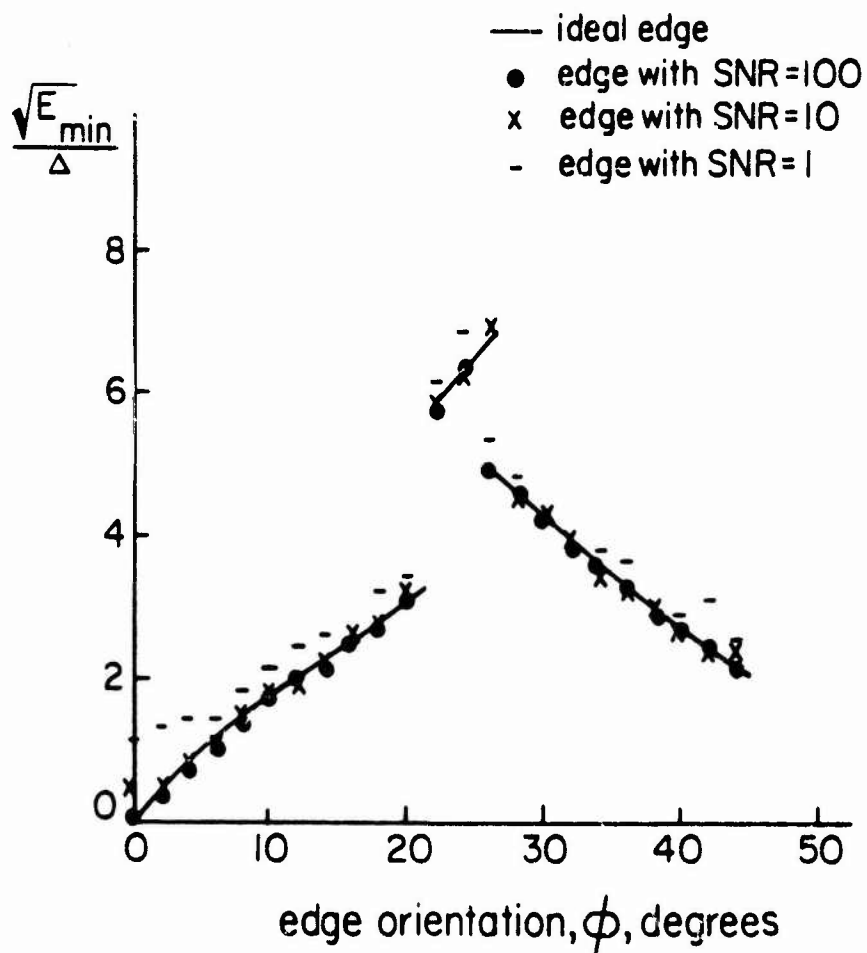
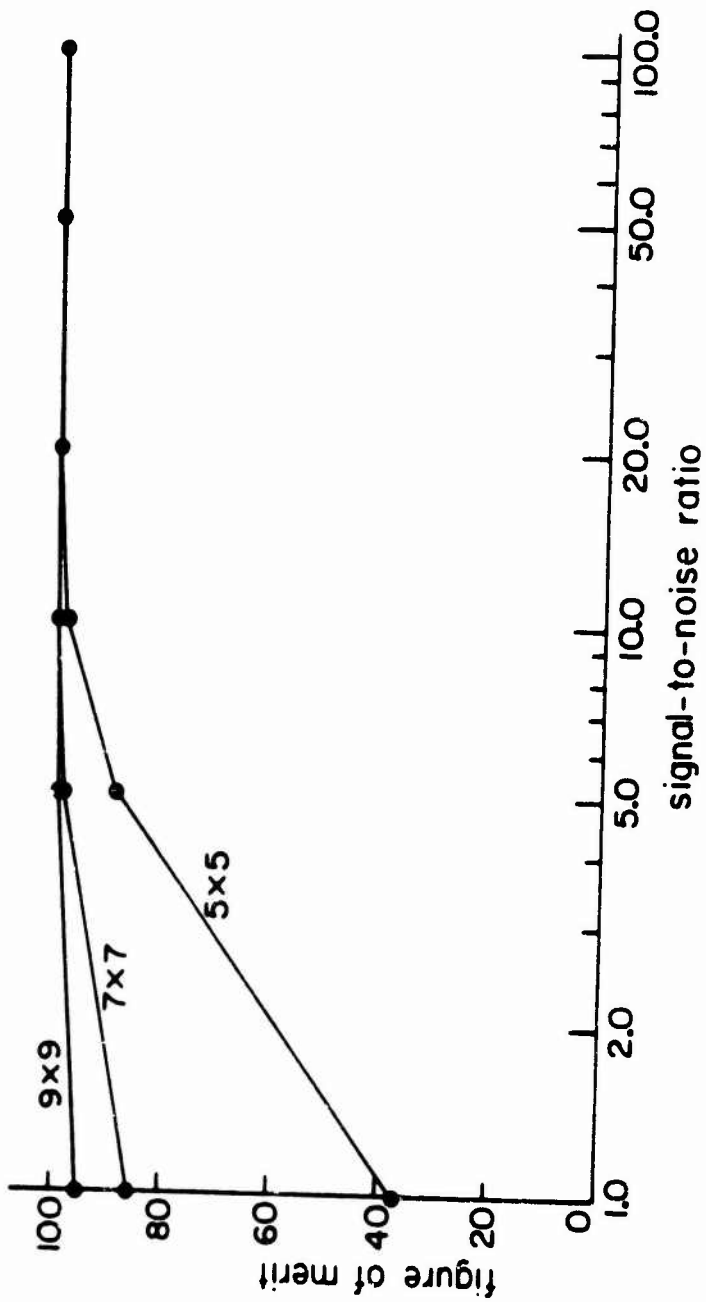


Figure 7.5. Edge fitting normalized error $\sqrt{E_{\min}}/\Delta$, as a function of actual edge orientation for noisy edges

Chapter 5. The results obtained, for the vertical and the diagonal test images, are shown in Figure 7.6. Examples of the edge maps for $SNR = 1.0$, are shown in Figure 7.7. Comparing the previous results with the results obtained for 3-level simple operators with the same mask sizes, it can be noticed that for small mask size and very low SNR, the edge fitting algorithm is not as good as the simple mask operators. This observation can be explained by the fact that the edge fitting algorithm bases its decision on an estimation of the edge parameters. This estimation is sensitive to noise especially when the number of pixels used is small. However, the edge fitting algorithm has better performance for high SNR and for large mask size. This is because the edge fitting algorithm suppresses displaced edges efficiently. The edge fitting algorithm has the additional advantage of being less sensitive to changes in the signal-to-noise ratio of the image. This results from using a decision strategy that is based on the normalized fitting error.

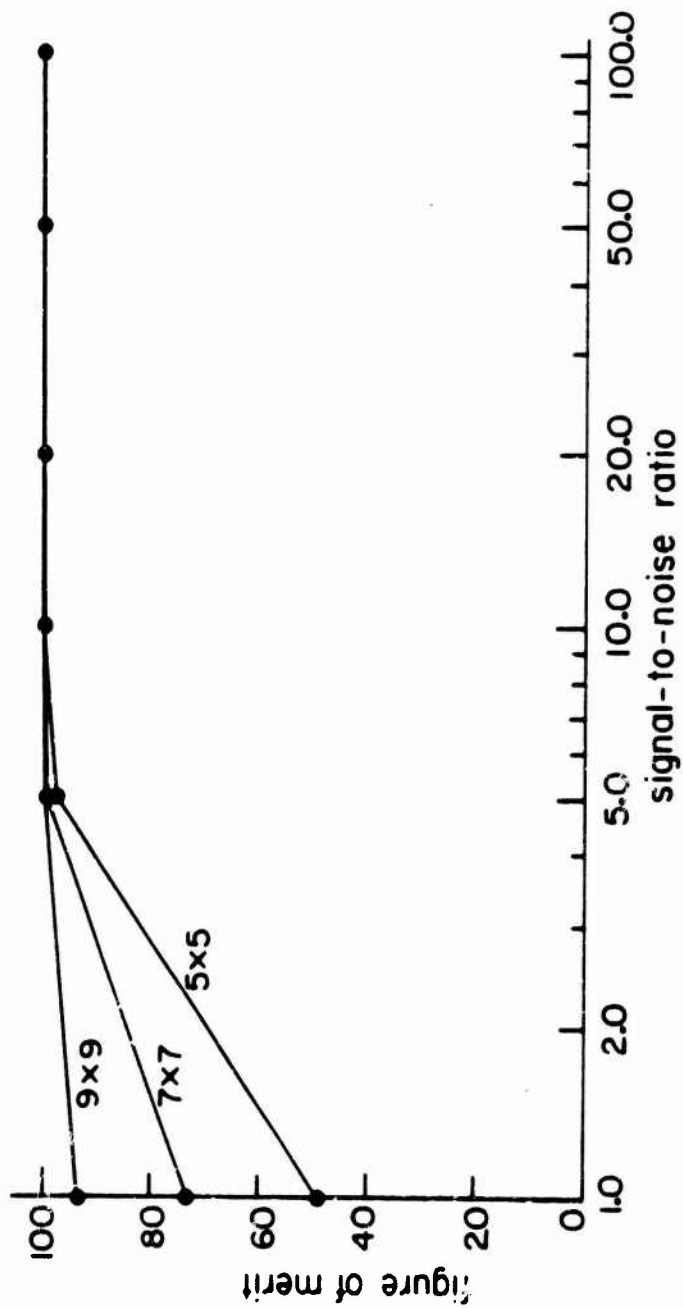
7.4 Conclusion

In this chapter, a new edge fitting algorithm has been introduced. The new algorithm is derived in the discrete domain, this allows a direct optimization of the operator's performance. The performance of the new algorithm is better than that of the edge enhancement/thresholding



a) vertical edge

Figure 7.6. Figure of merit as a function of signal-to-noise ratio for edge fitting operator



b) diagonal edge

Figure 7.6. (Continued)



a) original



b) 5x5 mask



c) 7x7 mask



d) 7x7 mask

Figure 7.7. Edge maps for the edge fitting operator, diagonal test image with SNR=1

operators for a wide range of signal-to-noise ratios.

Chapter 8

Conclusion and Further Work

This chapter summarizes the basic findings of the dissertation, and discusses the subjects that will need further investigation.

The objective of this work, was to introduce a quantitative analysis of the edge detectors, with an emphasis on the edge detectors as local operators, that can be used to preprocess the input images, without any a priori knowledge of the images contents. The tools that have been used in this analysis are the statistical detection theory and pattern classification. These concepts, help in a better understanding of the edge detection problem. Numerical ordering of the performance of the local edge detectors, was achieved by introducing a figure of merit defined for specific test images. New techniques for edge detection, including a discrete edge fitting algorithm, have been discussed.

There are, however, more questions to be answered before a complete understanding of the edge detection problem is achieved. First, in the case of the

differential edge detectors, the decision is based on measurements of the differences along two perpendicular axes. It is not clear, however, that combining these two differences in the sum of squares or the sum of magnitudes, is the optimum decision strategy. An optimum strategy can be developed if the probability density function of the edge orientation $p(\phi)$, is known.

Second, in all the previous analysis the edges are assumed to have specific orientations and heights. This is not true in real world images, where edges of various orientations and heights are present. The optimum threshold for this general case, can be derived if the statistical properties of the image is known.

Third, there is no efficient procedure to utilize the additional information about the edge height and orientation. Also, the best compromise between the mask size, the number of masks used, and the distance between consecutive applications of the edge detector, is not yet known.

The previous problems can be solved if a statistical image model is derived. This model will help in extending the techniques of this dissertation to the higher level of image understanding, such as edge linking and the recognition of image objects.

Appendix A

Analysis of the Hueckel Algorithm

Although the Hueckel algorithm possesses a theoretically optimum performance, there are two basic difficulties with the practical application of the operator. The first concerns the effect of truncation of the orthogonal expansion, while the second results from inaccuracies in the minimization procedure and in the computation of edge parameters. These two problems will be discussed in the following sections. In Section A-1, a review of the Hueckel algorithm is given. In Sections A.2 and A.3, the various difficulties with the algorithm are considered.

A.1 A Review of the Hueckel Algorithm

The Hueckel algorithm starts with the image intensities defined over a circular image subregion. A polar Fourier expansion of the image subregion is calculated, using the orthogonal functions given by Eqs. H.7* through H.8. The expansion is truncated to the first nine coefficients, a_0, a_1, \dots, a_8 . These coefficients

*This notation indicates equations in Hueckel's paper [14].

are compared with the ideal edge-line model coefficients, (s_0, s_1, \dots, s_8) . Expressions of s_i are given by Eqs. H.9 through H.10.

Acceptance of the edge fitting is based on three sequential decisions, each decision taken as soon as the information needed is available. The first decision is based on the inequality: If

$$\sum_{i=0}^8 a_i^2 < \frac{27}{64} \quad (\text{A.1})$$

then classify the subregion as no-edge. Equation A.1 discards the image subregions whose input amplitude varies less than that of a central edge of step height 1.5.

The error between the ideal and the actual signals can be expressed in the form

$$N = \frac{2}{3}a_1^2 + \frac{1}{2}a_2^2 + \frac{1}{2}a_3^2 + a_4^2 + a_5^2 + \frac{1}{2}a_6^2 + \frac{1}{2}a_7^2 - M(C_x, C_y) + f([a_i], p) \quad (\text{A.2})$$

where

$$M(C_x, C_y) = (e_2 C_x + e_3 C_y)^2 + e_4 C_x + e_5 C_y \quad (\text{A.3})$$

$$C_x = c_x^2 - c_y^2 \quad (\text{A.4})$$

$$C_y = 2c_x c_y \quad (\text{A.5})$$

The e_i 's are defined between Eqs. H.17 and H.19, while $f\{.,.\}$ corresponds to the last five terms in Eq. H.12. The vector p is the ideal edge parameter vector defined in Eq. 2.23. The best edge fitting is obtained by changing the parameter vector p , until N becomes minimum. Hueckel argued that at the minimum N , the function $f\{.,.\}$ vanishes. Hence to minimize N , it is sufficient to maximize $M(C_x, C_y)$ over C_x and C_y . The maximization of $M(C_x, C_y)$ is achieved, approximately, by Eqs. H.20 through H.21.

The signal power, $\sum s_i^2$, is evaluated, using the coefficients $[a_i]$ and the parameters C_x and C_y . Then, the second edge fitting decision is based on the criterion: Classify the subregion as no-edge if

$$2 \sum s_i^2 < \sum a_i^2 \quad (\text{A.6})$$

This inequality indicates that the noise power exceeds the signal power.

The parameters r_-, r_+, t_-, t_+ and b_- are calculated by Eqs. H.23 through H.25. These equations satisfy the condition of $f\{.,.\}$ being zero when N is minimum.

The final, and most important decision in the edge fitting procedure, is to compare the fitting error with a variable threshold. This is described by the criterion: If

$$\sum (a_i - s_i)^2 < \text{Conf}(\sum s_i^2) - \text{Diff} \quad (\text{A.7})$$

classify the subregion as an edge. The constant "Conf" relates to the edge distinctness and "Diff" relates to the edge pronouncedness. In evaluating Eq. A.7, different forms of $[s_i]$ are used for the three models, general edge-lines, edges, and lines, respectively.

The previous discussion reviewed the basic concepts of the Hueckel algorithm. It should be noticed that while the algorithm is theoretically optimum, it suffers from some deficiencies in its practical application. These deficiencies will be explained in the following sections.

A.2 Effect of Truncation of the Orthogonal Expansion

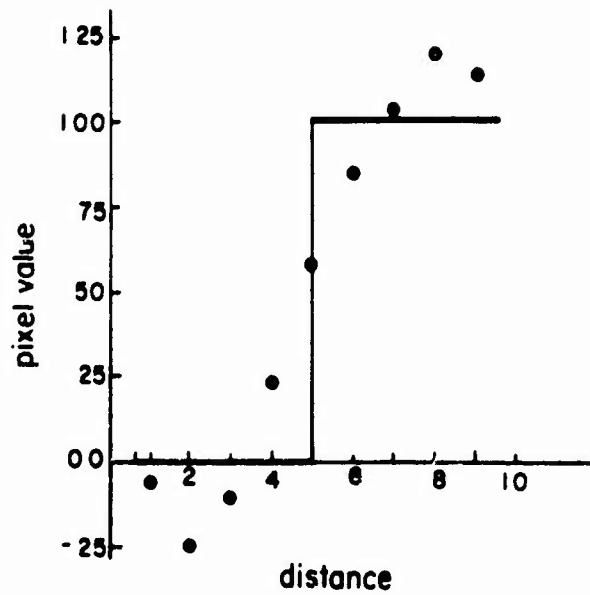
Hueckel assumed that the use of eight, and later of nine, coefficients of the orthogonal expansion will not affect the edge fitting performance because real edges are blurred and thus have small high spatial frequency components, while these high frequency components usually result from noise. This assumption is not true, especially if the subregion contains a line. To determine the effect of this approximation, the first nine Fourier coefficients of image subregions containing ideal edges and lines are calculated and then used to reconstruct the original signal. The original and reconstructed signals, in the case of ideal central edge and ideal lines of width 1 and

3, are given in Figure A.1. The results show the distortion introduced by truncation, especially in the case of thin lines.

The previous experiment leads to two questions: The first, what is the advantage of an optimum procedure if the models used are far from ideal? It should be noticed that the Hueckel algorithm suffers from difficulties in the detection of very thin lines [27]. This can be explained by the fact that the first nine coefficients of the Fourier expansion are not sufficient to represent thin lines accurately. The second question is, are the orthogonal functions chosen by Hueckel the best for the truncated expansion? This point is not important if an infinite expansion is used, as long as the orthogonal functions form a complete space. However, if a truncated expansion is used, it is important to choose orthogonal functions that are more sensitive to the ideal signals of interest. This is not the case in the Hueckel algorithm, where the functions H_i are chosen such that the optimization procedure can be solved analytically.

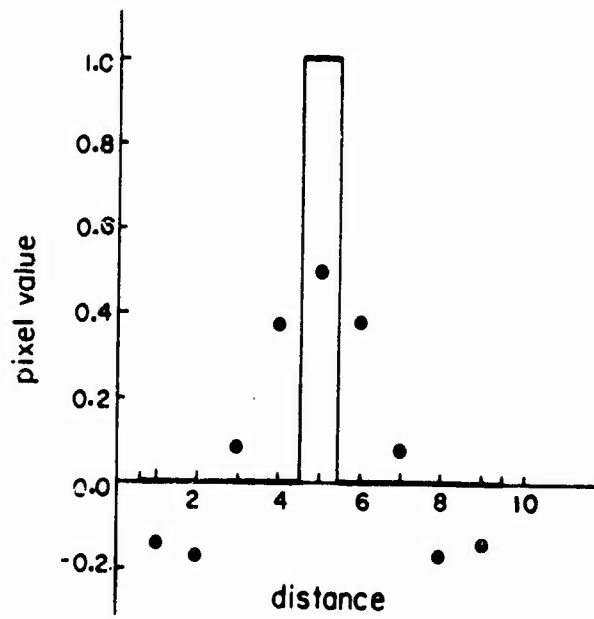
A.3 Effect of Inaccuracy in the Minimization Procedure.

The minimization procedure implemented by Hueckel suffers from difficulties that results in a suboptimum solution. These difficulties are summarized as follows:

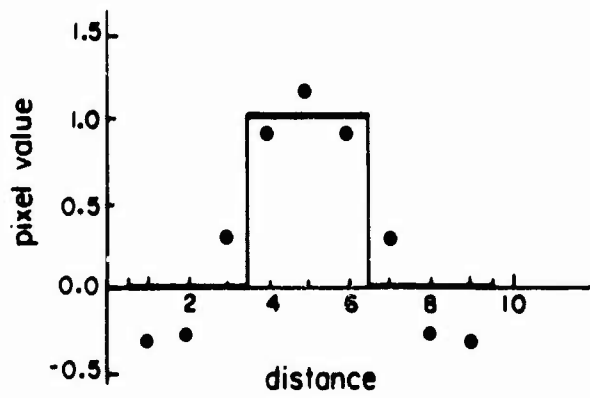


a) central edge

Figure A.1. Ideal and reconstructed edge and line models. — ideal, • reconstructed



b) line, width = 1



c) line, width = 3

Figure A.1. (Continued)

First, in the minimization procedure the parameter vector (p) is allowed to assume complex values and also to indicate edges with centers outside the circular subregion. Although the previous two conditions do not represent acceptable solutions, Hueckel has to allow these generalized form to simplify the algorithm. The parameters are readjusted by neglecting the imaginary parts, and ignoring the edges whose centers are outside the circular subregion. It is clear that this solution will not be the same as the optimum solution obtained with the previous constraints taken into consideration.

Second, the replacement of the minimization of Eq. A.2 by the maximization of Eq. A.3 is based on the assumption that the minimum of $f\{[a_i], p\}$ is zero. This assumption is valid only if p is real [28], which is not true in general. In fact, for the terms of $f\{[a_i], p\}$ to vanish, the following equations should be satisfied

$$b_1(c_x, c_y) = \lambda_+ + \lambda_- \quad (\text{A.8a})$$

$$b_2(c_x, c_y) = \lambda_+ r_+ + \lambda_- r_- \quad (\text{A.8b})$$

$$b_3(c_x, c_y) = \lambda_+ r_+^2 + \lambda_- r_-^2 \quad (\text{A.8c})$$

$$b_4(c_x, c_y) = \lambda_+ r_+^3 + \lambda_- r_-^3 \quad (\text{A.8d})$$

where the b_i 's are functions of the set $\{a_i\}$ and the parameters c_x and c_y , while λ_{\pm} are defined as

$$\lambda_{\pm} = t_{\pm} (3\pi)^{\frac{1}{2}} (1-r_{\pm}^2)^2 / 4 \quad (\text{A.9})$$

It is clear that the solution of Eq. A.8 will, in general, result in complex values of λ_+ , λ_- , r_+ , r_- . A real solution will be guaranteed if and only if the image subregion corresponds to an ideal edge model.

Third, in arranging the terms in Eq. H.12, s_g is artificially set equal to a_g . This assumption cannot be justified. As a result of this constraint, the accuracy of the second Hueckel algorithm [14], is not expected to be better than the accuracy of his first algorithm [8]. It seems that s_g was made equal to a_g only to simplify the minimization procedure.

A quantitative evaluation of the effect of the previous approximations on the Hueckel operator performance would be quite involved. Such an evaluation is not attempted here. Instead, an experimental evaluation of the operator's performance is given. In the experiment, the Hueckel operator is applied on the vertical test image introduced in Chapter 5. The figure of merit is plotted as a function of signal-to-noise ratio for different choices of Hueckel's parameters, Conf and Diff. These plots are shown in Figure A.2. It can be noticed that the

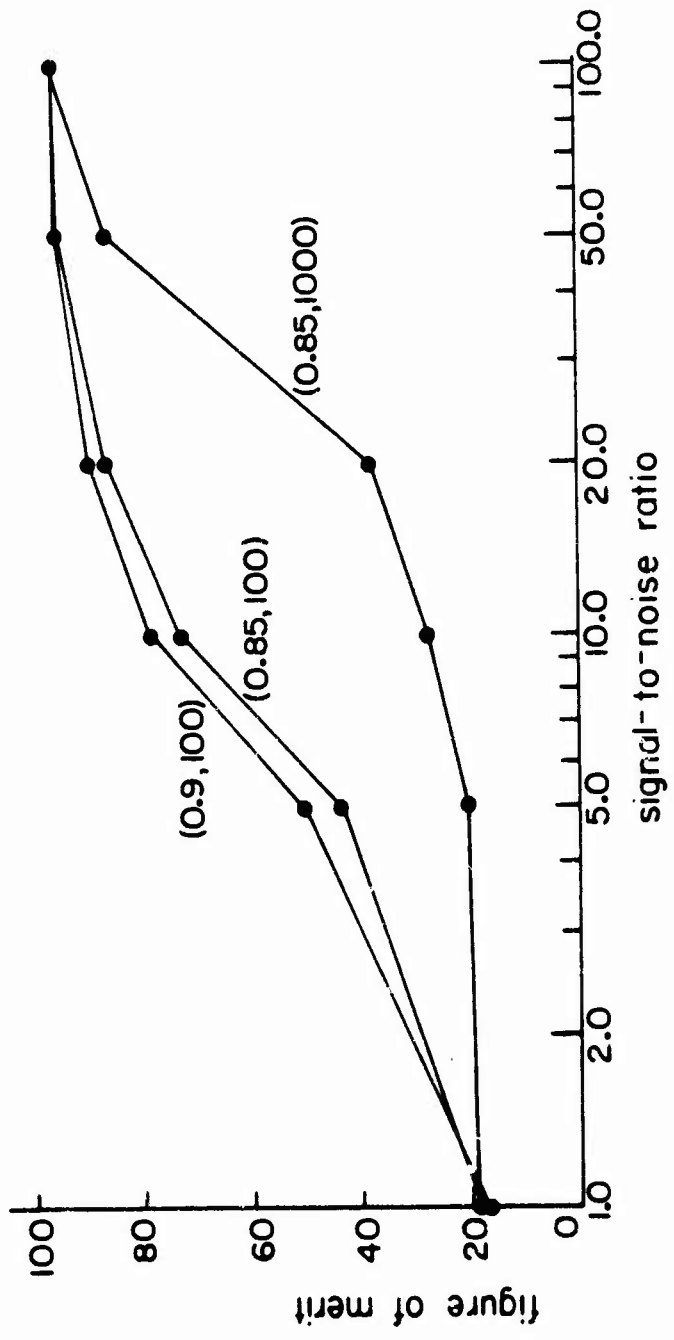


Figure A.2. Figure of merit as a function of signal-to-noise ratio for the Hueckel operator

performance of the Hueckel operator is inferior to that of the simple operators given in Chapter 6, and it is also inferior to the edge fitting algorithm introduced in Chapter 7.

Appendix B

Orthogonal Transformation in Edge Detection

One of the early applications of orthogonal transformation in edge detection was given by Hueckel in his edge fitting algorithm [8,14]. The method implements a truncated polar Fourier expansion in the fitting procedure. Later, a simplified version of the Hueckel algorithm was introduced by Mero and Vassy [29]. In this procedure, only two of the Fourier components are used in the edge detection. This simplification results in unacceptable loss in performance when detecting roisy edges [30].

In the Hueckel algorithm, the orthogonal transformation was used to simplify the edge fitting procedure. A different application of the orthogonal transformation is to use it as a multidimensional rotation of the feature space [31]. This approach can be useful if the edge and no-edge features are enhanced by the transformation. The following sections discuss this new approach. In Sections B.1 and B.2, calculations of the Fourier components of different edge and line models are given. In section B.3, a preliminary analysis of the performance of this new technique is introduced.

B.1 Edge models in the Discrete Fourier Domain

The edge model in the spatial domain is sketched in Figure B.1. The edges are assumed to have one of the four basic orientations: vertical, horizontal, positive slope diagonal, and negative slope diagonal. Central edges are considered first, and then the analysis is extended to non-central edges. If the edge is described by the function $f(j,k)$, where $-N \leq j, k \leq N$, the corresponding Fourier coefficients $F(u,v)$ are defined as

$$F(u,v) = \frac{1}{(2N+1)^2} \sum_{k=-N}^N \sum_{j=-N}^N f(j,k) w^{ju+kv} \quad (\text{B.1})$$

where

$$w = \exp\left(\frac{2\pi i}{2N+1}\right) \quad (\text{B.2})$$

In many cases, the corresponding discrete Fourier coefficients can be derived in closed forms. As an example, in the case of the central vertical edge shown in Figure B.1b, the Fourier coefficients are of the form

$$F_V(0,0) = b + \frac{h}{2} \quad (\text{B.3})$$

otherwise

$$F_V(u,0) = \frac{1}{2N+1} \left[\frac{1}{2} + \frac{w^{(N+1)u} - w^u}{w^u - 1} \right] \quad (\text{B.4})$$

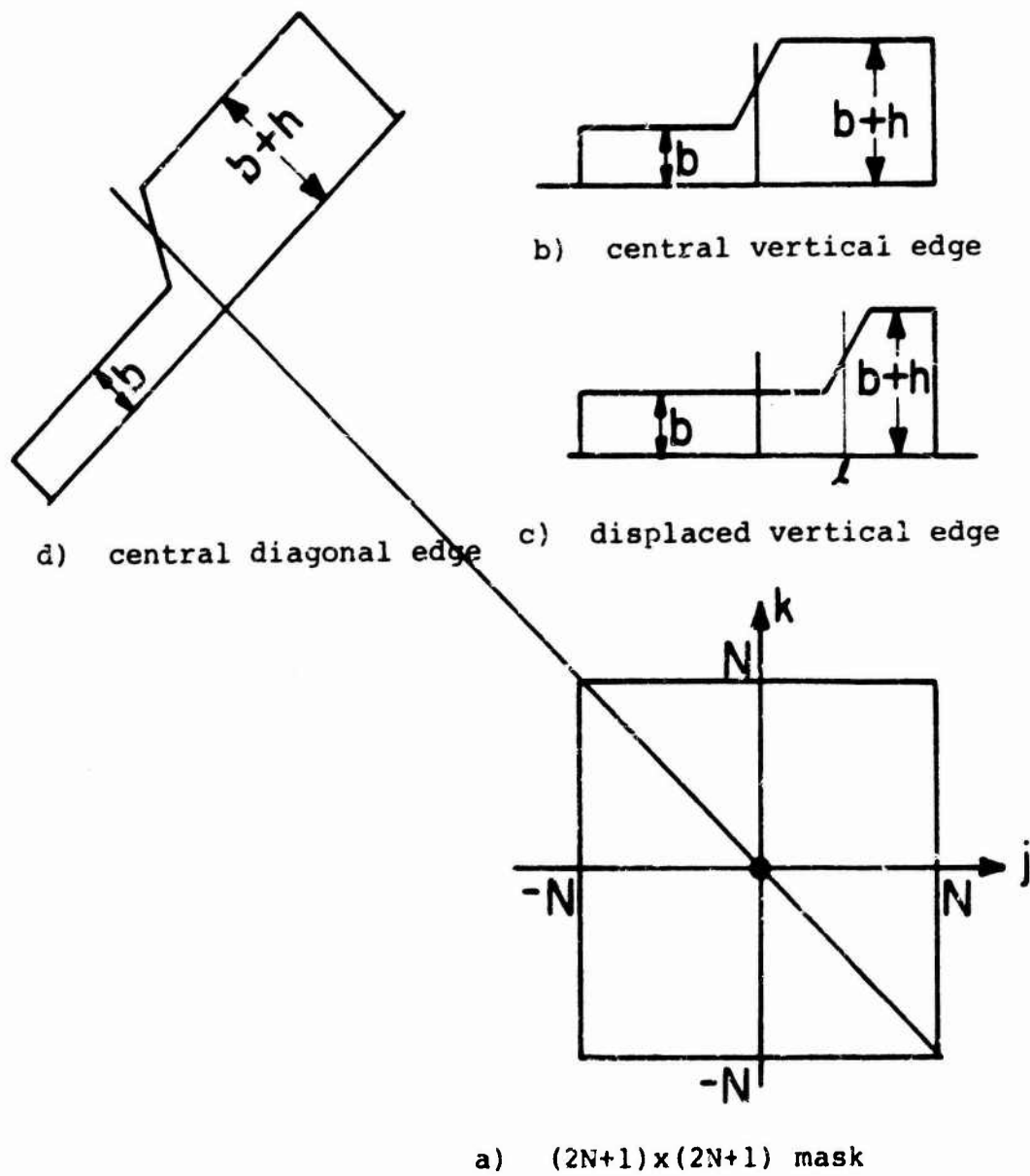


Figure B.1. Edge models for the discrete Fourier transform

$$F_v(u, v) = 0 \quad v \neq 0 \quad (\text{B.5})$$

In the case of the central diagonal edge shown in Figure B.1d, the Fourier coefficients are

$$F_{\pi/4}(0, 0) = b + \frac{h}{2} \quad (\text{B.6})$$

otherwise

$$F_{\pi/4}(u, 0) = -\frac{h}{(2N+1)} \frac{w^{Nu}}{w^{-u}-1} \quad u \neq 0 \quad (\text{B.7})$$

$$F_{\pi/4}(0, v) = \frac{h}{(2N+1)} \frac{w^{-Nv}}{w^v-1} \quad v \neq 0 \quad (\text{B.8})$$

$$F_{\pi/4}(u, v) = 0 \quad u, v \neq 0 \quad (\text{B.9})$$

Similar expressions can be obtained for edges with $\phi = \pi/2$ and $\phi = 3\pi/4$.

The previous analysis can be extended to the case of noncentral edges and edges with general orientation. To avoid repetition, only one of these general cases is considered. This is the case of the displaced vertical edge shown in Figure B.1c. The corresponding discrete Fourier components are given by

$$F_l(0, 0) = b + \frac{h}{2} - \frac{hl}{(2N+1)} \quad (\text{B.10})$$

otherwise

$$F_{\ell}(u,v) = \frac{h}{(2N+1)} w^{\ell u} \left[\frac{1}{2} + \frac{w^{(N+1-\ell)u} - w^u}{w^u - 1} \right] \quad (\text{B.11})$$

$$F_{\ell}(u,v) = 0 \quad v \neq 0 \quad (\text{B.12})$$

The discrete Fourier coefficients in the case of a 5x5 central edge with different orientations are calculated. Results are tabulated in Figure B.2. From these results, it is clear that edge orientation can be determined from the Fourier coefficients. A decision strategy based on these Fourier coefficients will be given in Section B.3.

B.2 Line Models in the Discrete Fourier Domain

Line detection was excluded from this dissertation for two reasons. First, lines can be detected as two consecutive edges, especially if the edge detector used, possesses small masks. Second, template matching line detectors suffer from the problem of being very sensitive to the line orientation and position, and so far, it seems there is no practical solution to this problem. It is hoped that the sensitivity problem can be avoided by using the discrete Fourier transformation. This approach will be introduced in the following paragraphs.

Figure B.3 shows discrete models for one-pixel-width

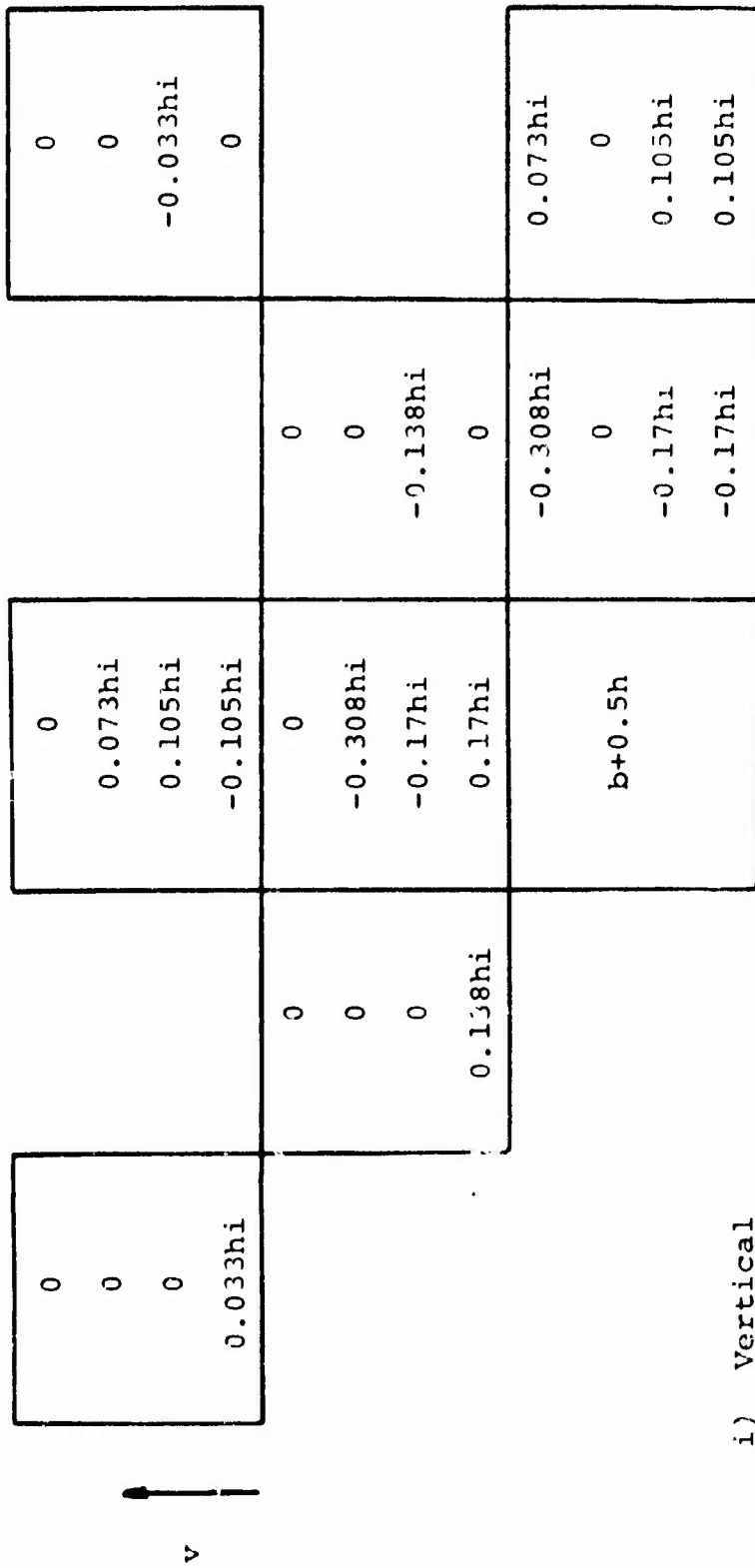


Figure B.2. Fourier Coefficients for a 5x5 edge

b	b	b	b+h	b	b	b
b	b	b	b+h	b	b	b
.
.
.
b	b	b	b+h	b	b	b
b	b	b	b+h	b	b	b

a) Vertical line

b+	b+	b	b	b	b	b
0.914h	0.25h					
b+	b+	b+	b	.	.	b
0.25h	0.914h	0.25h				
b	.	.	.			b
b		.	.	.		b
b			.	.	.	b
b				.	.	b+
						0.25h
b	b	b	b	b	b+	b+
					0.25h	0.914h

b) Diagonal line

Figure B.3. A one-pixel-line model

lines with vertical and diagonal orientations. The vertical line has the Fourier coefficients

$$F_V(0,0) = b + \frac{h}{(2N+1)} \quad (B.13)$$

otherwise

$$F_V(u,0) = \frac{h}{(2N+1)} \quad (B.14)$$

$$F_V(u,v) = 0 \quad v \neq 0 \quad (B.15)$$

The diagonal line has the Fourier coefficients

$$F_{\pi/4}(0,0) = b + \frac{h}{(2N+1)} \left[0.914 + \frac{N}{(2N+1)} \right] \quad (B.16)$$

otherwise

$$F_{\pi/4}(u,u) = \frac{h}{(2N+1)} \left[0.914 + \frac{N}{(2N+1)} \cos\left(\frac{2\pi u}{2N+1}\right) \right] \quad (B.17)$$

$$F_{\pi/4}(u,-u) = \frac{-h}{2(2N+1)^2} (-1)^u \quad (B.18)$$

$$F_{\pi/4}(u,v) = \frac{-h}{4(2N+1)^2} w^{-N(-u+v)} (w^u + w^{-v}) \quad (B.19)$$

In the case of a vertical line at a distance l from the origin in the spatial plane, the discrete Fourier components become

$$F_{\rho}(0,0) = b + \frac{h}{2N+1} \quad (\text{B.20})$$

otherwise

$$F_{\rho}(u,0) = \frac{hw^{\rho}u}{2N+1} \quad (\text{B.21})$$

$$F_{\rho}(u,v) = 0 \quad v \neq 0 \quad (\text{B.22})$$

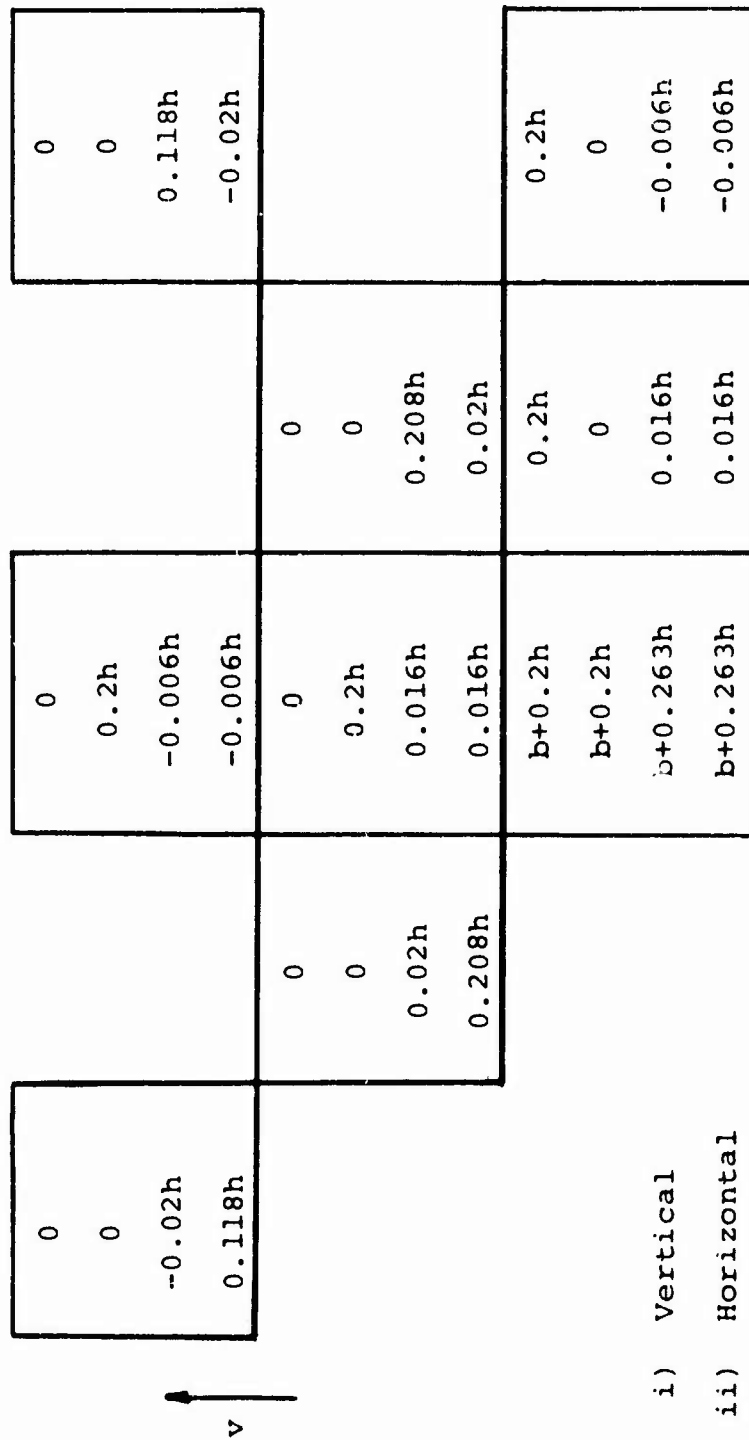
It should be noticed that the only difference between the Fourier components of a central and a displaced vertical line is a phase factor in $F_{\rho}(u,0)$. The changes are more pronounced in the case of shifted diagonal lines.

The Fourier coefficients of a 5x5 central line with different orientations are calculated and results are tabulated in Figure B.4. Again it is clear that the line orientation can be determined from the Fourier coefficients.

To show the improvement in sensitivity that can be achieved by using the orthogonal transformation, the discrete Fourier coefficient of the rotated line shown in Figure B.5a are computed. In this case, $F_{\rho}(u,v)$ is given by

$$F_{\rho}(0,0) = b + \frac{h}{5} \quad (\text{B.23})$$

otherwise



- i) Vertical
- ii) Horizontal
- iii) $\pi/4$
- iv) $3\pi/4$

Figure B.4. Fourier coefficients for a 5x5 line

b	b+0.8h	b+0.2h	b	b
b	b+0.4h	b+0.6h	b	b
b	b+0.05h	b+0.9h	b+0.05h	b
b	b	b+0.6h	b+0.4h	b
b	b	b+0.2h	b+0.8h	b

a) Rotated vertical line

-1	-1	+4	-1	-1
-1	-1	+4	-1	-1
-1	-1	+4	-1	-1
-1	-1	+4	-1	-1
-1	-1	+4	-1	-1

b) Vertical line template

Figure B.5. Detection of a rotated vertical line

$$F_r(u,v) = \frac{h}{25} \left[0.9 + 1.2 \cos\left(\frac{2\pi v}{5}\right) + 0.4 \cos\left(\frac{4\pi v}{5}\right) + 0.1 \cos\left(\frac{2\pi v}{5}\right) + 0.8 \cos\left(\frac{2\pi(u-v)}{5}\right) + 1.6 \cos\left(\frac{2\pi(u-2v)}{5}\right) \right] \quad (\text{B.24})$$

The value of $F_r(1,0)$ is

$$F_r(1,0) = 0.1309h \quad (\text{B.25})$$

This represents a ratio of 0.65 of the value $F_v(1,0)$. On the other hand, if the template matching operator, shown in Figure B.5b, is used, the output in the case of the rotated line will be

$$X_r = 7.5h \quad (\text{B.26})$$

This represents a ratio of 0.375 of the value X_v .

B.3 Performance Analysis of the Discrete Fourier Transform Edge Detector

The performance of the previous edge and line detectors can be evaluated using the statistical model of Chapter 3. In this model, the spatial function $f(j,k)$ is the sum of a signal and a noise component

$$f(j,k) = \tilde{f}(j,k) + n(j,k) \quad (\text{B.27})$$

where $n(j,k)$ is an additive white Gaussian noise with zero mean and standard deviation σ . The corresponding discrete Fourier coefficients $F(u,v)$ will be, in general, complex random variables. The real and imaginary components of

$F(u,v)$ can be arranged in the vector form

$$\underline{F} = \begin{bmatrix} F_R(-N,N) \\ F_I(-N,N) \\ \vdots \\ F_R(N,-N) \\ F_I(N,-N) \end{bmatrix} \quad (\text{B.28})$$

where \underline{F} is a joint Gaussian vector with mean

$$\underline{\tilde{F}} = \frac{1}{(2N+1)^2} \begin{bmatrix} \sum_j \sum_k \tilde{f}(j,k) \cos \left[\frac{2\pi(-Nj+Nk)}{2N+1} \right] \\ \vdots \\ - \sum_j \sum_k \tilde{f}(j,k) \sin \left[\frac{2\pi(Nj-Nk)}{2N+1} \right] \end{bmatrix} \quad (\text{B.29})$$

and the covariance matrix is

$$\underline{\Sigma} = \frac{\sigma^2}{2(2N+1)^2} \mathbf{I} \quad (\text{B.30})$$

In Eqs . B2 8 to B.30, the term corresponding to $F(0,0)$ is excluded. Therefore, the identity matrix \mathbf{I} is of size $2[(2N+1)^2-1]$.

The fact that the different components of \underline{F} are independent Gaussian simplifies the performance evaluation. As an example, if the decision strategy is to detect a vertical edge when

$$|F_I(1,0)| \geq t_1 \quad (\text{B.31})$$

the probability of correct detection of a vertical edge is

$P(\text{vertical edge}|\text{vertical edge}) =$

$$1 - \text{erf} \left[\frac{\sqrt{2}(2N+1)}{\sigma} (t_1 - 0.308h) \right] - \text{erf} \left[\frac{\sqrt{2}(2N+1)}{\sigma} (t_1 + 0.308h) \right] \quad (\text{B.32})$$

and the probability of false detection is

$$P(\text{vertical edge}|\text{no-edge}) = 1 - 2\text{erf} \left[\frac{\sqrt{2}(2N+1)}{\sigma} t_1 \right] \quad (\text{B.33})$$

Similar expressions can be obtained for $P(\text{vertical edge}|\text{horizontal edge})$ and $P(\text{vertical edge}|\text{diagonal edge})$. The threshold t_1 can be chosen to satisfy a required probability of false detection while maximizing the probability of correct detection.

Better performance can be achieved, however, if edge detection is based on simultaneous comparison of the Fourier coefficients. Thus, edge detection becomes multiple hypotheses testing in a vector space. This approach needs further investigation.

Appendix C

Derivations of Eqs. 3.29, 3.31 and 3.32

In deriving these equations, it should be noticed that the equation

$$A = |X| + |Y| \tag{C.1}$$

corresponds to lines 1, 2, 3 and 4 in Figure C.1. Thus the probability density function $p(A)$ is given by

$$\begin{aligned}
 p(A) = & \int_{Y=0}^A p_X(A-Y)p_Y(Y)dY + \int_{Y=0}^A p_X(Y-A)p_Y(Y)dY \\
 & + \int_{Y=-A}^0 p_X(-A-Y)p_Y(Y)dY + \int_{Y=-A}^0 p_X(A+Y)p_Y(Y)dY
 \end{aligned} \tag{C.2}$$

and

$$\begin{aligned}
 P(A < t) = & \int_{Y=-t}^0 \int_{X=-Y}^Y p_X(X)p_Y(Y)dXdY \\
 & + \int_{Y=0}^t \int_{X=-Y}^Y p_X(X)p_Y(X)dY
 \end{aligned} \tag{C.3}$$

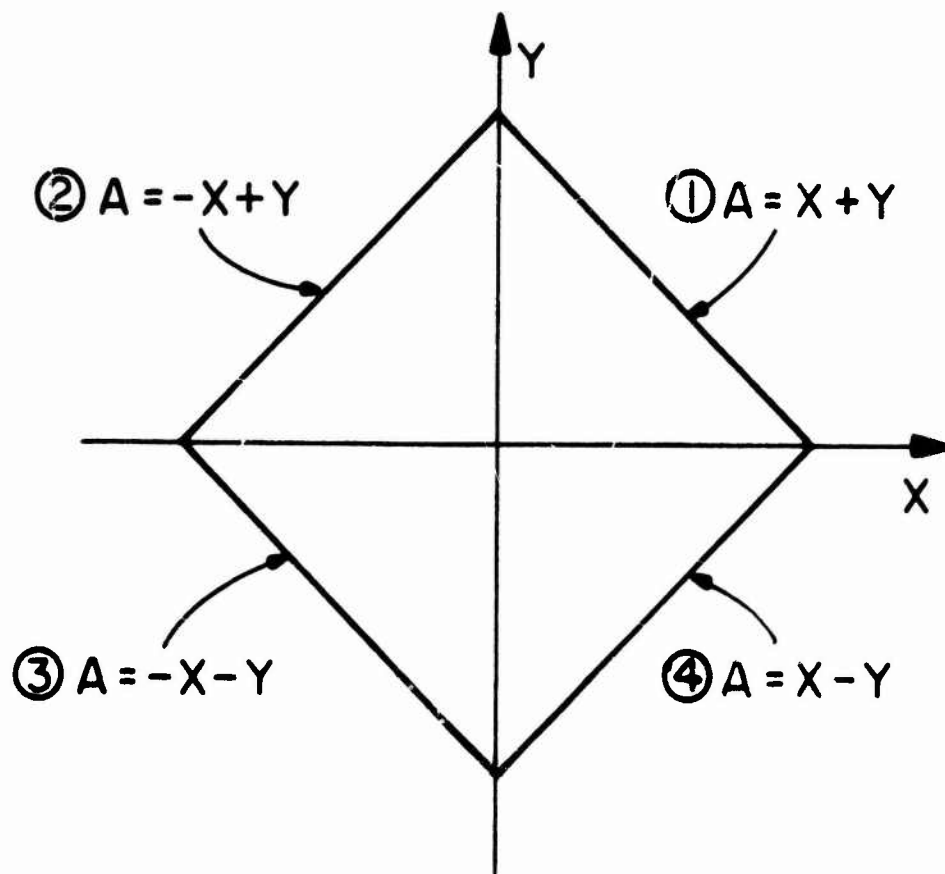


Figure C.1. The equation $A = |X| + |Y|$

Appendix D

The Herskovits Algorithm

Concepts of statistical detection theory were first utilized in the design of edge detectors by Griffith [9], Yakimovsky [10], and Herskovits [26]. A brief discussion of Griffith and Yakimovsky techniques, was given in Chapter 1. In this appendix, a discussion of the Herskovits approach, and its resemblance to the statistical analysis of Chapter 3 is given.

Herskovits was interested in processing images that contain polyhedra. The edges of a polyhedron can be in the form of ideal or defocussed steps and roofs. These intensity models should be distinguished from the unwanted signals that take the form of constant slow slopes and Gaussian noise.

The Herskovits algorithm computes the second difference at every point. This is given by

$$\begin{aligned} D(x) &= [f(x+\delta) - f(x)] - [f(x) - f(x-\delta)] \\ &= -2f(x) + f(x+\delta) + f(x-\delta) \end{aligned} \tag{D.1}$$

where $f(x)$ is the intensity function shown in Figure 7.1,

and δ is a fixed interval. A two sided cutoff (α) is put on $D(x)$ so that if $|D(x)| < \alpha$, then $D(x)$ is set to 0. Next, the function $F_s(x)$ is computed as

$$F_s(x) = \sum_{i=1}^{\delta} \text{sg}(D(x+i)) - \sum_{i=1}^{\delta} \text{sg}(D(x-i)) \quad (\text{D.2})$$

where

$$\text{sg}(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases} \quad (\text{D.3})$$

Actually, $F_s(x)$ is computed over a two-dimensional neighborhood. Finally, local maxima of $F_s(x)$ are found, and a line fitting procedure builds the complete edge [32].

The edge detector parameters were chosen to maximize the probability of correct detection for a given probability of false detection. This approach resembles the statistical analysis introduced in Chapter 3. The basic differences between the Herskovits technique and the analysis of this dissertation can be summarized in the following.

First, Herskovits was interested in a limited domain of images. Thus, the class of edges and no-edges were determined by a priori knowledge of the image contents and the imaging process. This kind of knowledge was not implemented in the present dissertation.

Second, in the analysis given by Herskovits, edges were assumed to be vertical. To detect other edge orientations, the operators should be rotated. This assumption simplified the derivation of a statistical model, but limited its application. The analysis given in Chapters 2 and 3 of this dissertation is based on a general edge model, that has been used in evaluating the performance of different edge detectors.

Third, Herskovits was attempting to achieve an almost error free detection because the systems used to recognize polyhedra are very sensitive to errors introduced in the low levels of image processing. It seems that a better strategy of image understanding systems should allow for larger probability of error at the low levels, that can be improved later by feedback from the high levels of image processing.

Appendix E
Experimental Results

The models used in edge detectors evaluation assume that images consist of ideal steps or ramps affected by additive white Gaussian noise. In real world pictures, however, noise is often considered to be the irrelevant image intensities such as the background. It is important to determine edge detector performance for both artificial and actual image models.

A simple procedure to achieve this comparison is to test the different edge detectors using real world pictures. Examples of this experiment are shown in Figures E.1, E.2 and E.3. In these examples, the 3x3 Prewitt operator, the 3x3 and 7x7 3-level operator, the 7x7 edge fitting algorithm and the Hueckel operator are applied on test pictures containing a girl, an airport and a tank. The thresholds for the Hueckel and the edge fitting algorithms are fixed at optimum values; $Conf = 0.85$, $Diff = 100$, for Hueckel and $t = .045$ for the edge fitting algorithm. The thresholds for the Prewitt and the 3-level operators are chosen so that the number of edges detected equals the number of edges detected by the Hueckel



a) original



b) 3x3 mask, Prewitt operator



c) 3x3 mask, 3-level operator



d) 7x7 mask, 3-level operator

Figure E.1. Examples of edge maps, girl picture



e) 7x7 mask, edge
fitting operator

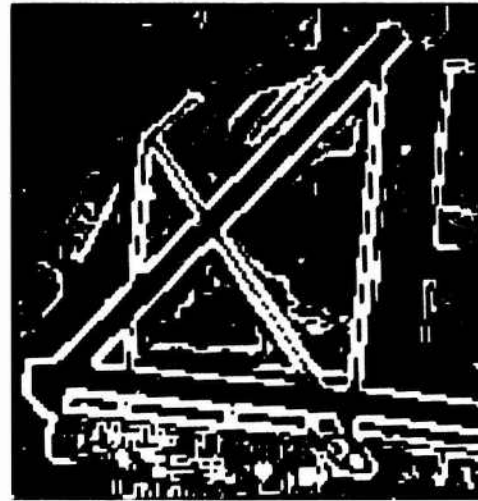


f) 69 pixels, Hueckel
operator

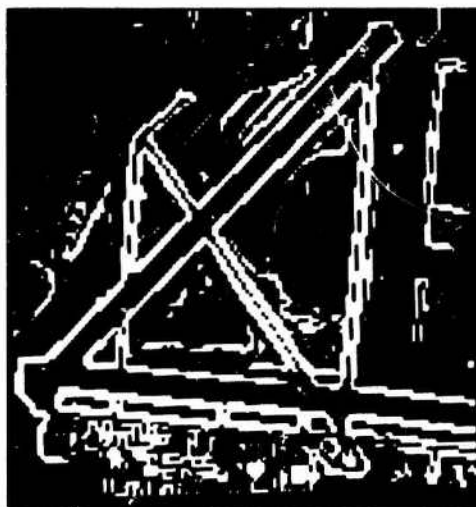
Figure E.1. (Continued)



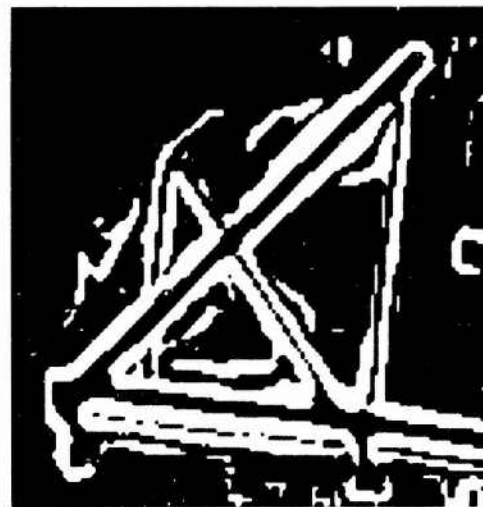
a) original



b) 3x3 mask, Prewitt operator



c) 3x3 mask, 3-level operator



d) 7x7 mask, 3-level operator

Figure E.2. Examples of edge maps, airport picture

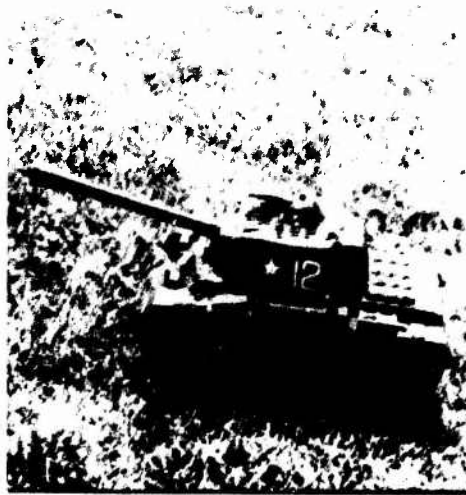


e) 7x7 mask, edge fitting operator



f) 69 pixels, Hueckel operator

Figure E.2. (Continued)



a) original



b) 3x3 mask, Prewitt operator



c) 3x3 mask, 3-level operator

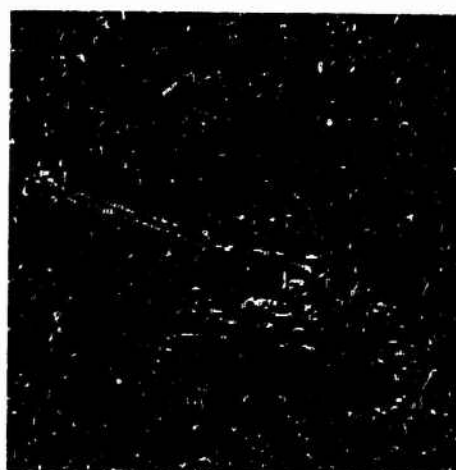


d) 7x7 mask, 3-level operator

Figure E.3. Examples of edge maps, tank picture



e) 7x7 mask, edge
fitting operator



f) 6° pixels, Hueckel
operator

Figure E.3. (Continued)

algorithm.

In comparing the performance of the edge enhancement/thresholding operators with that of the edge fitting algorithms, it is seen that the edge fitting algorithms are better able to outline the "usually" relevant scene content. This results from the more general edge models used in the edge fitting algorithms, that allow for detection of out-of focus objects. Also, it should be observed that while the edge fitting algorithms use fixed thresholds, the thresholds of the edge enhancement/thresholding operators have to be varied for different images.

For the edge enhancement/thresholding operators, the 3x3 Prewitt and 3-level operators have practically the same performance. Also, the effects of increasing the mask size, namely, suppression of noise and lowering the operator resolution, are apparent in the tank pictures.

The new edge fitting algorithm has better performance than that of the Hueckel operator because the new algorithm preserves more of the relevant structure of the pictures.

These observations have been predicted previously in the dissertation. This shows that there is a correlation between the artificial and actual image models. Further investigation of this assumption, based on quantitative

measurements is still needed.

References

1. W.K. Pratt, Digital Image Processing, Wiley-Interscience, New York, 1978.
2. L.G. Roberts, "Machine Perception of Three-Dimensional Solids," in Optical and Electro-Optical Information Processing, J.T. Tippett et al., Eds., MIT Press, Cambridge, Mass., 1965, pp. 159-197.
3. J.M.S. Prewitt, "Object Enhancement and Extraction," in Picture Processing and Psychopictorics, B.S. Lipkin and A. Rosenfeld, Eds., Academic Press, New York, 1970.
4. R. Duda, and P. Hart, Pattern Classification and Scene Analysis, John Wiley, New York, 1973.
5. A. Rosenfeld and A.C. Kak, Digital Picture Processing, Academic Press, New York, 1976.
6. R.A. Kirsch, "Computer Determination of the Constituent Structure of Biological Images," Computers and Biomedical Research, Vol.4, No.3, June 1971, pp. 315-328.
7. G.S. Robinson, "Edge Detection by Compass Gradient Masks," Note, Computer Graphics and Image Processing, Vol.6, No.5, October 1977, pp. 492-501.

8. M.H. Hueckel, "An Operator Which Locates Edges in Digitized Pictures," J. Assoc. Comput. Mach., Vol.18, No.1, January 1971, pp. 113-125.
9. A.K. Griffith, "Mathematical Models for Automatic Line Detection," J. Assoc. Comput. Mach., Vol.20, No.1, January 1973, pp. 62-80.
10. Y. Yakimovsky, "Boundary and Object Detection in Real World Images," Proc. of the 4th Int. Joint Conference on Artificial Intelligence, Moscow, Sept. 1975, pp. 695-704.
11. M. Kelly, "Edge Detection by Computer Using Planning," in Machine Intelligence VI, Edinburgh Univ. Press, Edinburgh, 1971, pp. 397-409.
12. C.K. Chow and T. Kaneko, "Boundary Detection of Radiographic Images by a Threshold Method," in Frontiers of Pattern Recognition, S. Watanabe, Ed., Academic Press, New York, 1972, pp. 61-82.
13. J.R. Fram, and E.S. Deutsch, "On the Quantitative Evaluation of Edge Detection Schemes and Their Comparison with Human Performance," IEEE Trans. on Computers, Vol. C-24, No.6, June 1975, pp. 616-628.
14. M.H. Hueckel, "A Local Visual Operator Which Recognizes Edges and Lines," J. Assoc. Comput. Mach., Vol.20, No.4, October 1973, pp. 634-647.

15. H.L. Van Trees, Detection, Estimation, and Modulation Theory-Part I, John Wiley, New York, 1968.
16. K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, New York, 1972.
17. K.S. Miller, Multidimensional Gaussian Distribution, John Wiley, New York, 1964.
18. M. Schwartz, W.R. Bennett, and S. Stein, Communication Systems and Techniques, McGraw Hill, New York, 1966.
19. D. Middleton, An Introduction to Statistical Communication Theory, McGraw Hill, New York, 1960.
20. J.T. Tou, and R.C. Gonzalez, Pattern Recognition Principles, Addison-Wesley, 1974.
21. Y.C. Ho and R.L. Kashyap, "An Algorithm for Linear Inequalities and its Applications," IEEE Transactions on Electronic Computers, Vol.EC-14, No.5, October 1965, pp. 683-688.
22. F.W. Smith, "Pattern Classifier Design by Linear Programming," IEEE Trans. on Computers, Vol. C-17, No.4, April 1968, pp. 361-372.
23. E. Argyle, "Techniques for Edge Detection," Proceedings IEEE, Vol.59, No. 2, February 1971, pp. 285-287 (letter).

24. I.D.G. Macleod, "Comments on Techniques for Edge Detection," Proceedings IEEE, Vol.60, No.3, March 1972, pp. 344.
25. A. Rosenfeld, M. Thurston, and Y.H. Lee, "Edge and Curve Detection: Further Experiments," IEEE Trans. on Computers, Vol. C-21, No. 7, July 1972, pp. 677-715.
26. A. Herskovits, "On Boundary Detection," Proj. MAC Artificial Intelligence Memo 183, MIT, Cambridge, Mass., 1970.
27. R. Nevatia, private communication.
28. G.B. Shaw, "Local and Regional Edge Detectors: Some Comparisons," University of Maryland Technical Report TR-614, December 1977.
29. L. Mero and Z. Vassy, "A Simplified and Fast Version of the Hueckel Operator for Finding Optimal Edges in Pictures," Proc. of the 4th Int. Joint Conference on Artificial Intelligence, Moscow, Sept. 1975, pp. 650-655.
30. R. Nevatia, "Evaluation of a Simplified Hueckel Edge-Line Detector," Note, Computer Graphics and Image Processing, Vol.6, No.6, December 1977, pp. 582-588.
31. H.C. Andrews, Mathematical Techniques in Pattern Recognition, Wiley-Interscience, New York, 1972.

32. L.S. Davis, "A Survey of Edge Detection Techniques,"
Computer Graphics and Image Processing, Vol. 4, No.3,
September 1975, pp. 248-270.