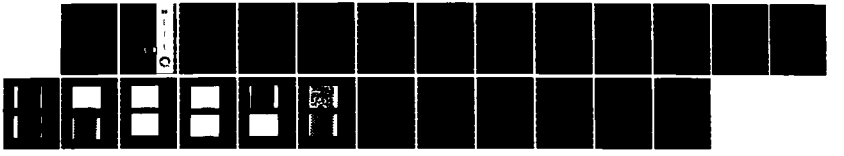
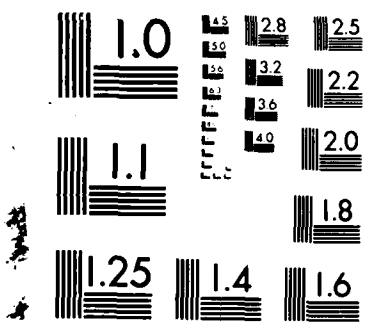
AD-A173 883 AN EDGE DETECTION EXPERIMENT USING THE MARR OPERATOR (U) ARMY ENGINEER TOPOGRAPHIC LABS FORT BELVOIR VA M A CROMBIE ET AL. AUG 86 ETL-0435 1/1

UNCLASSIFIED F/G 9/4 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)

ETL - 0435

An edge detection experiment using the MARR operator

Michael A. Crombie
Edward H. Bosch

August 1986

Prepared for
U.S. ARMY CORP OF ENGINEERS
ENGINEER TOPOGRAPHIC LABORATORIES
FORT BELVOIR, VIRGINIA 22060-5546

E T L

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: ETL-0435	2. GOVT ACCESSION NO.: ADA173883
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE *(and Subtitle)*: AN EDGE DETECTION EXPERIMENT USING THE MARR OPERATOR	5. TYPE OF REPORT & PERIOD COVERED
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Michael A. Crombie Edward H. Bosch	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: U.S. Army Engineer topographic Laboratories Fort Belvoir, VA 22060-5546	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 4A762707A855
11. CONTROLLING OFFICE NAME AND ADDRESS: U.S. Army Engineer Topographic Laboratories Fort Belvoir, VA 22060-5546	12. REPORT DATE: August 1986 13. NUMBER OF PAGES: 20
14. MONITORING AGENCY NAME & ADDRESS*(If different from Controlling Office)*	15. SECURITY CLASS. *(of this report)*: Unclassified 15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution is unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, If different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

An edge detection experiment using the MARR operator was conducted at ETL to determine if the operator could be used in an interactive manner to extract well-defined edges from noisy pictures. Three binary image-processing functions, namely, line thinning, line thickening and clutter removal were found to be useful aids in the interactive process. The results of the experiment demonstrate that the MARR operator was not biased and performed as well as the Prewitt and Haralick operators, all of which is contrary to an experiment conducted by Haralick where the operator was used incorrectly.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

PREFACE

This study was conducted under DA Project 4A762707A855, Topographic Mapping Technology.

The study was conducted during 1985 under the supervision of Mr. Dale E. Howell, Chief, Information Sciences Division, and Mr. Lawrence A. Gambino, Director, Computer Sciences Laboratory.

COL Alan L. Laubscher, CE, was Commander and Director, and Mr. Walter E. Boge was Technical Director during the report preparation.

CONTENTS

	Page
PREFACE	i
ILLUSTRATIONS	iii
TABLES	iv
INTRODUCTION	1
NUMERICAL EXPERIMENT	2
Performance Measures	2
Checkerboard #1 - Noise Model #1	2
Checkerboard #1 - Noise Model #2	5
Checkerboard #2	5
DISCUSSION	5
CONCLUSIONS	13
APPENDIXES	
A. Marr Edge Operator	14
B. Checkerboard Models	15
C. Image Processing Functions	16

ILLUSTRATIONS

Figure No.	Title	Page
1	Checkerboard #1 - σN = 10	6
2	Checkerboard #1 - σN = 30	6
3	Checkerboard #1 - σN = 50	7
4	Edge Window (σN = 10) - σM = 5	7
5	Edge Window (σN = 30) - σM = 5	8
6	Edge Window (σN = 50) - σM = 5	8
7	Processed Edge Window (σN = 10, σM = 5)	9
8	Processed Edge Window (σN = 30, σM = 5)	9
9	Processed Edge Window (σN = 50, σM = 5)	10
10	Edge Window (σN = 50) - σM = 2	10
11	Checkerboard #1 - Noise Model #2	11
12	Checkerboard #2	11
13	Checkerboard Edge Definition	12

TABLES

Table No.	Title	Page
1	Checkerboard #1 - Noise Model #1 - $^{\sigma}M = 2$	2
2	Checkerboard #1 - Noise Model #1 - $^{\sigma}M = 3$	3
3	Checkerboard #1 - Noise Model #1 - $^{\sigma}M = 4$	3
4	Checkerboard #1 - Noise Model #1 - $^{\sigma}M = 5$	3
5	$(^{\sigma}M = 5) \bullet$ AND $\bullet (^{\sigma}M = 2)$	4
6	$(^{\sigma}M = 5) \bullet$ AND $\bullet (^{\sigma}M = 3)$	4
7	$(^{\sigma}M = 5) \bullet$ AND $\bullet (^{\sigma}M = 4)$	4
8	Checkerboard #1 - Noise Model #2	5
9	Checkerboard #2	5

INTRODUCTION

Reference is made to the war of words between R. Haralick on one hand and W. Grimson and E. Hildreth on the other.[1,2,3] The discussion centered around determining step edges from noisy digital pictures wherein Haralick purported to prove that theMarr edge detector did not do as well as the Prewitt operator and neither did as well as the Directional Derivative edge operator developed by Haralick. Grimson and Hildreth point out that Haralick made improper use of the Marr operator, an observation with which we agree.

The Marr operator was developed as a model for mammalian visual systems where a raw primal sketch of an image is developed numerically by calculating edge pictures at several resolutions. The full primal sketch is constructed by grouping the raw primitives in a variety of ways.[4] A brief description of the Marr operator and our approach to computing the edge picture is given in appendix A.

Haralick suggests that the digital picture should be regarded as a representation of the true image where random noise has been added to each pixel. We suggest that this is a far too simple representation of the imaging event. The real difficulty stems from clutter, where sought-for detail is obscured by other real world detail. Since no edge finder can locate hidden edges without a knowledge base to fill in the blanks, we are more concerned at this time in using edge finders in an interactive mode to detect real world lines and edges. One of the first developments along these lines was demonstrated at ETL on the DIAL system. In that experiment, a drainage pattern was extracted interactively from a digital image.[5] Essentially, we operated in a computer-aided mode where the Marr operator extracted the drainage pattern along with a host of unwanted, yet real world, edges. Image processing functions, which are described in appendix C, were used interactively to extract the drainage pattern from the unwanted detail.

The purpose of the work described in this Research Note is to demonstrate that the Marr edge finder compares favorably to the Prewitt and to Haralicks's, especially when simple image processing functions are applied to the edge picture. The images used in this work were similar to the noisy checkerboards used by both authors. Since most edge finders work well when the edge is apparent, reasonably well when only faintly obscured and not at all when the edge is totally obscurred, we feel that the argument as to the best algorithm may be similar to a tempest in a teapot.

[1] R. Haralick, "Digital Step Edges from Zero Crossing of Second Directional Derivatives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 1, January 1984.

[2] W. Grimson and E. Hildreth, "Comments on Digital Step Edges from Zero Crossings of Second Directional Derivatives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No. 1, January 1985.

[3] R. Haralick, "Author's Reply," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No. 1, January 1985.

[4] D. Marr and E. Hildreth, *Theory of Edge Detection*, Massachusetts Institute of Technology Artificial Intelligence Laboratory, AI MEMO No. 518, April 1979.

[5] N. Friend, *Analysis of Interactive Image Cleansing Via Raster-Processing Techniques*, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, VA, ETL-0347, November 1983, AD-A141 772.

NUMERICAL EXPERIMENT

Performance Measures

Three criteria were used to judge the performance of the Marr operator. The first criterion (P_1) is the conditional probability of an assigned edge given that the pixel was in fact a true edge. The second criterion (P_2) is the conditional probability of a true edge given that the process assigned an edge value to the pixel.

P_1 = Prob [Marr Edge | True Edge]
P_2 = Prob [True Edge | Marr Edge]

These creteria are identical to the probability measures used by Haralick and Grimson and Hildreth.

Haralick defined his error distance as "the average distance to closest true edge pixel of pixels which are assigned nonedge but which are true edge pixels." A straightforward interpretation of this definition is that the average error, and indeed any such error, is zero. We define the error distance as the average distance to the closest true edge pixel or pixels which are assigned to be edge pixels.

E = NER / NEQ

NEQ = Number of Marr Edges

NER = Sum of Marr Error Distances

Checkerboard #1 - Noise Model #1

A number of simulation runs as described in appendix B were run on the DIAL system. Three of the resulting noisy checkerboards are described in figures 1, 2, and 3. The five noisy scenes (σN = 10, 20, 30, 40, 50) were processed using four Marr resolution windows (σM = 2, 3, 4, 5). The resulting edge pictures were further processed using the raster cleansing algorithms described in appendix C. The performance criteria for the four Marr resolutions are presented in Tables 1, 2, 3, and 4. Note that each set of results are presented in two groups, namely, those results (#1) where thickening and thinning were applied and those results (#2) where they were not applied.

Table 1. Checkerboard #1 - Noise Model #1 - σM = 2

			#1			#2		
σN	CL	LIK = LTN	P_1	P_2	E	P_1	P_2	E
10	25	1	1.000	1.000	0.00	1.000	0.984	0.02
20	29	1	1.000	0.982	0.02	0.999	0.998	0.00
30	29	1	0.979	0.892	0.70	0.928	0.899	0.71

Table 2. Checkerboard #1 - Noise Model #1 - $\sigma M = 3$

			#1			#2		
σN	CL	LIK = LTN	P_1	P_2	E	P_1	P_2	E
10	19	1	1.000	0.984	0.02	0.999	1.000	0.00
20	23	1	1.000	0.980	0.03	0.992	0.994	0.01
30	25	2	0.987	0.884	0.74	0.911	0.895	0.71
40	25	2	0.952	0.789	1.53	0.822	0.796	1.50

Table 3. Checkerboard #1 - Noise Model #1 - $\sigma M = 4$

			#1			#2		
σN	CL	LIK = LTN	P_1	P_2	E	P_1	P_2	E
10	13	1	1.000	0.983	0.02	0.997	0.998	0.00
20	17	2	1.000	0.976	0.04	0.985	0.989	0.03
30	19	2	0.969	0.880	0.75	0.897	0.886	0.78
40	18	2	0.912	0.793	1.49	0.803	0.790	1.53
50	19	2	0.842	0.712	2.05	0.720	0.704	2.14

Table 4. Checkerboard #1 - Noise Model #1 - $\sigma M = 5$

			#1			#2		
σN	CL	LIK = LTN	P_1	P_2	E	P_1	P_2	E
10	0	1	1.000	0.981	0.02	0.993	0.996	0.00
20	11	2	1.000	0.972	0.06	0.978	0.984	0.05
30	11	2	0.964	0.877	0.73	0.882	0.882	0.76
40	11	2	0.888	0.789	1.42	0.792	0.787	1.46
50	13	2	0.814	0.709	2.04	0.712	0.705	2.11

In order to get a visual description of the process, compare figures 4, 5 and 6 to figures 1, 2 and 3. Figures 4, 5 and 6 are the resulting edge pictures when the Marr resolution window ($\sigma M = 5$) is applied to figures 1, 2 and 3. When the raster processing exercise is applied to figures 4, 5 and 6, the results can be seen in figures 7, 8 and 9.

Note the amount of noise in figure 3 when $\sigma N = 50$. Consider figure 10, which is the resultant edge window derived from figure 3, when the Marr resolution was $\sigma M = 2$. This figure demonstrates that low resolution values of the Marr operator act as high pass filters. In this case, the resultant edge image contained too many noise edges for raster processing to be effective. This is the reason tables 1 and 2 are incomplete.

Edge pictures at several resolutions were regarded by Marr and Hildreth as a raw primal sketch of the edge scene. A more complete primal sketch may be obtained by various groupings of the raw sketches. A sensible combination of the raw sketches should produce a more complete sketch of the edge scene and in our case, a more accurate representation. In order to test this notion we combined several of the processed results in pairs by declaring an edge at a pixel if each member of the pair also declared the pixel to be an edge. The results are presented in tables 5, 6 and 7. The headings #1 and #2 have the same meaning as before.

Table 5. (σM = 5) • AND • (σM = 2)

	#1			#2		
σ_N	P_1	P_2	E	P_1	P_2	E
10	1.000	0.984	0.02	0.993	1.000	0.00
20	1.000	0.983	0.02	0.977	0.999	0.00
30	0.944	0.965	0.14	0.829	0.978	0.13

Table 6. (σM = 5) • AND • (σM = 3)

	#1			#2		
σ_N	P_1	P_2	E	P_1	P_2	E
10	1.000	0.984	0.02	0.993	1.000	0.00
20	1.000	0.981	0.02	0.977	0.996	0.00
30	0.956	0.937	0.30	0.836	0.949	0.31
40	0.868	0.869	0.89	0.706	0.877	0.87

Table 7. (σM = 5) • AND • (σM = 4)

	#1			#2		
σ_N	P_1	P_2	E	P_1	P_2	E
10	1.000	0.983	0.02	0.993	0.998	0.00
20	1.000	0.977	0.04	0.977	0.991	0.02
30	0.952	0.905	0.56	0.856	0.912	0.59
40	0.870	0.825	1.20	0.748	0.825	1.26
50	0.785	0.745	1.80	0.655	0.740	1.90

Checkerboard #1 - Noise Model #2

In this example, the average gray shade of the 16 X 16 = 256 squares remains exactly as in the first example. The standard deviation of the noise was allowed to vary for each of the squares as described in appendix B. The resultant noisy checkerboard is shown in figure 11. It was processed using three Marr resolutions (σM = 5, 4 and 3). The resulting edge pictures were further processed using the raster cleansing algorithms and the results are presented in Table 8.

Table 8. Checkerboard #1 - Noise Model #2

σM	CL	LIK = LTN	P_1	P_2	E
3	25	1	0.948	0.886	0.72
4	19	1	0.933	0.882	0.72
5	9	2	0.953	0.876	0.68

Checkerboard #2

In this example, the average gray shade of the 256 squares varied as did the standard deviation of each square. The random model is described in appendix B. The resultant noisy checkerboard is shown in figure 12. It was processed using two Marr resolutions (σM = 5 and 4). The resulting edge pictures were further processed using the raster cleansing algorithms and the results are presented in table 9.

Table 9. Checkerboard #2

σM	CL	LIK = LTN	P_1	P_2	E
4	29	0	0.379	0.250	7.88
5	29	0	0.317	0.265	7.37
5	29	2	0.342	0.243	7.60

Discussion

The primary reason for this work was to show that the Marr operator, when properly used, compares favorably with the Prewitt and Haralick operators. This does not mean we endorse the Marr operator over all others. In fact, it seems at this time that more effort should be expended by researchers toward developing knowledge bases for filling in obscured edges than tweaking the multitude of edge operators, most of which are comparable in performance.

Haralick reported that the Marr operator is in some cases biased and that it and other edge operators must produce edges 2 pixels wide. The latter statement depends on how the edge is defined. We defined an edge in such a way that it occurred at specific pixels. Consider figure 13 below, which depicts a portion of a checkerboard where the squares are 4 by 4.

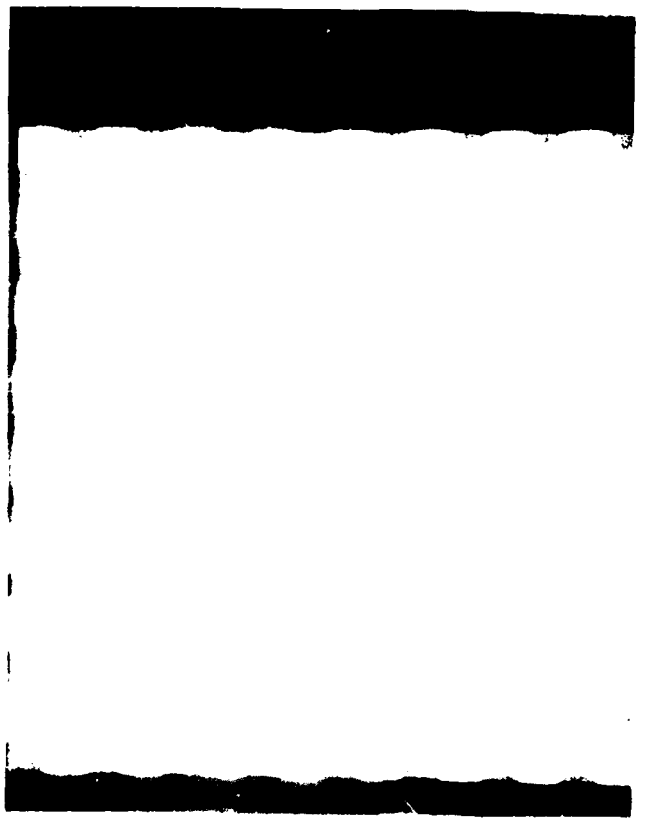


FIGURE 1. Checkerboard #1 $^{O}N = 10$

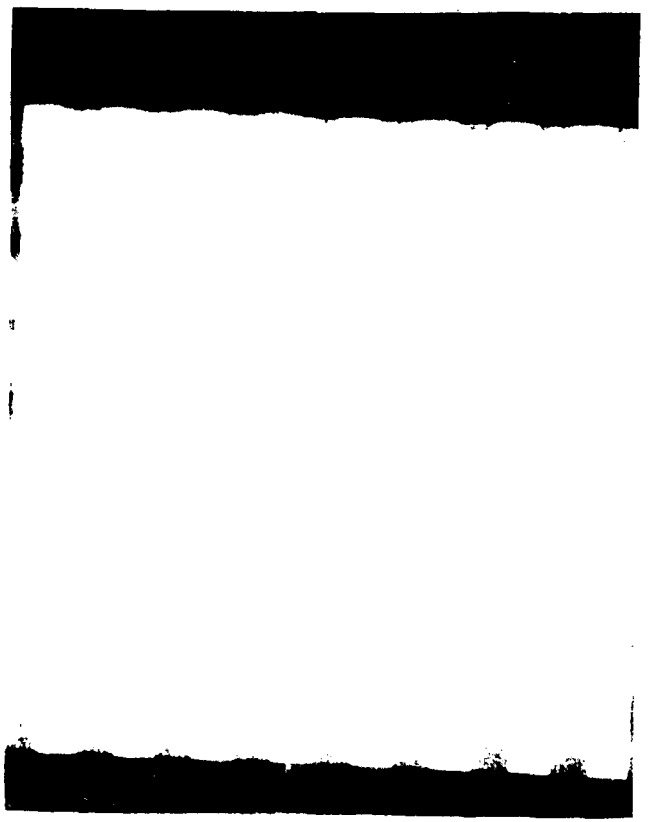


FIGURE 2. Checkerboard #1 $^{O}N = 30$

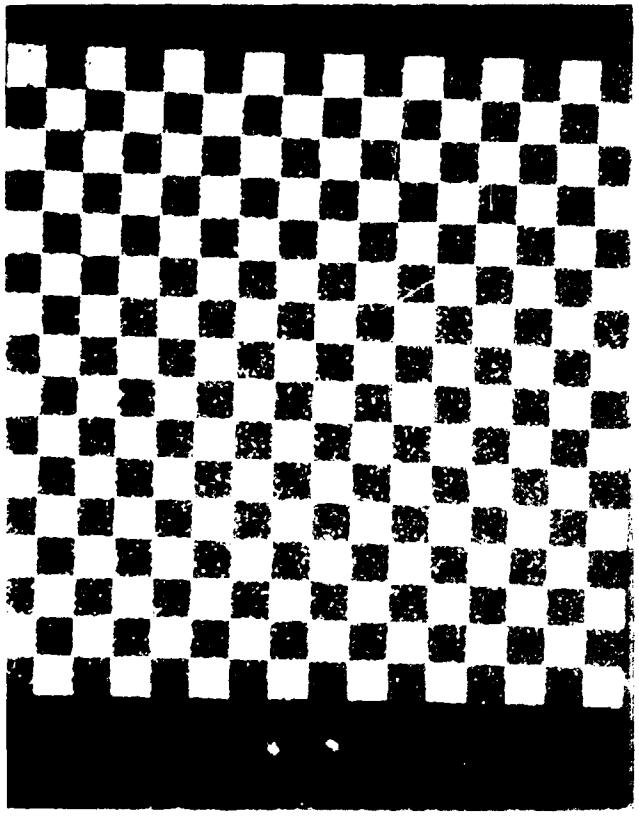


FIGURE 3. Checkerboard #1 $^{O}N = 50$

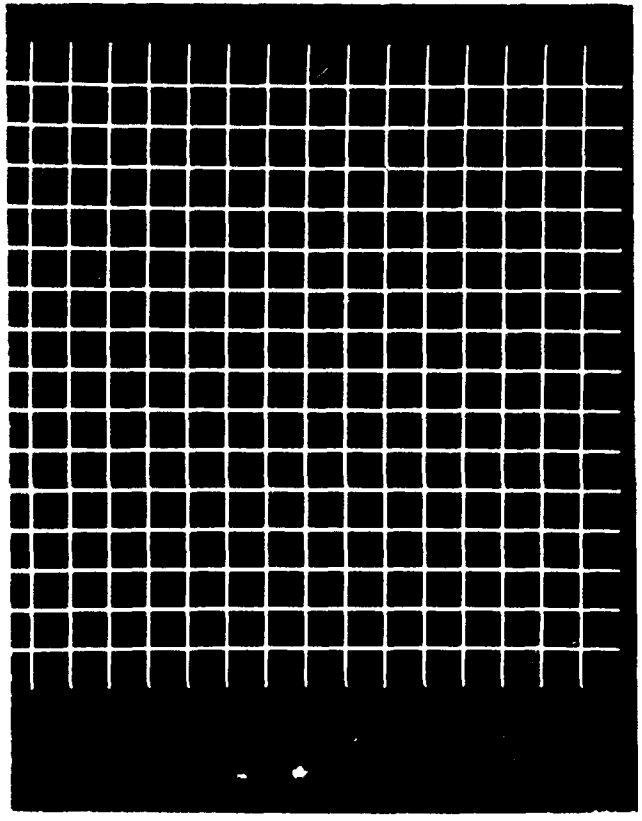


FIGURE 4. Edge Window ($^{O}N = 10$) — $^{O}M = 5$

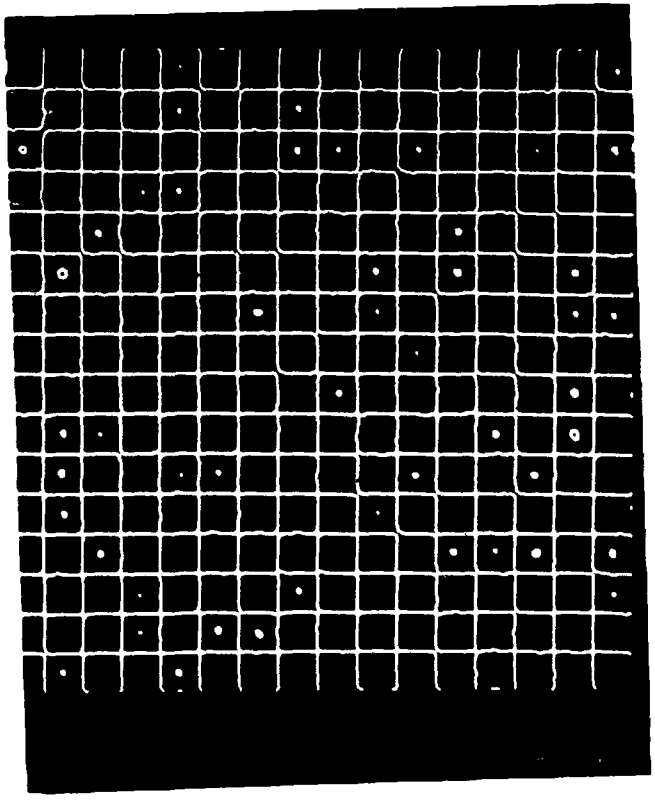


FIGURE 5. Edge window ($^{\sigma}N = 30$) - $^{\sigma}M = 5$

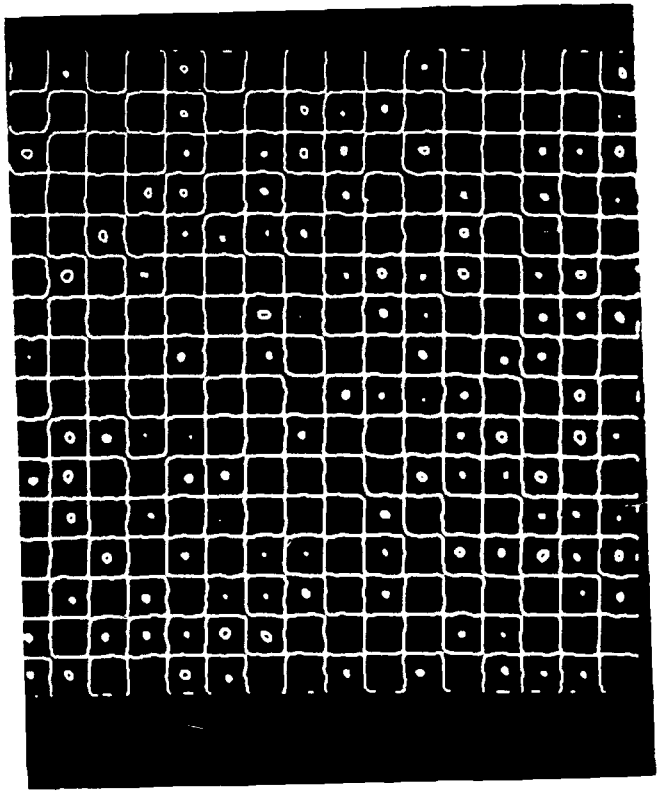


FIGURE 6. Edge window ($^{\sigma}N = 50$) - $^{\sigma}M = 5$

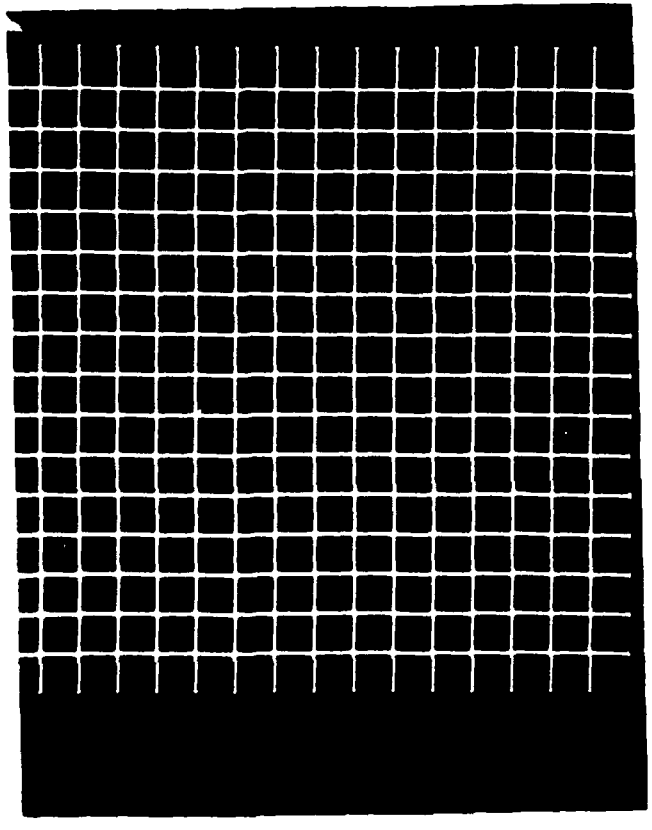


FIGURE 7. Processed edge window ($^{O}N = 10$, $^{O}M = 5$)

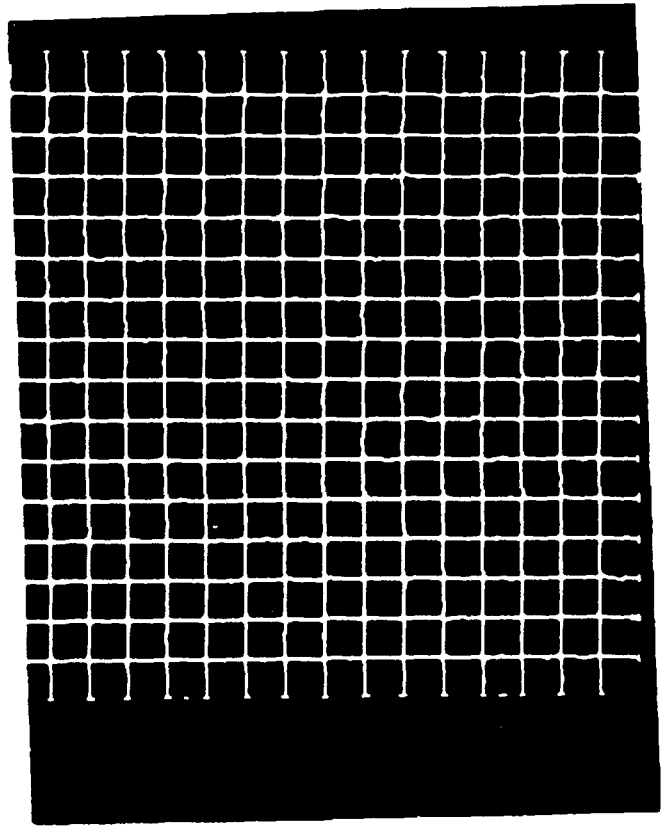


FIGURE 8. Processed edge window ($^{O}N = 30$, $^{O}M = 5$)

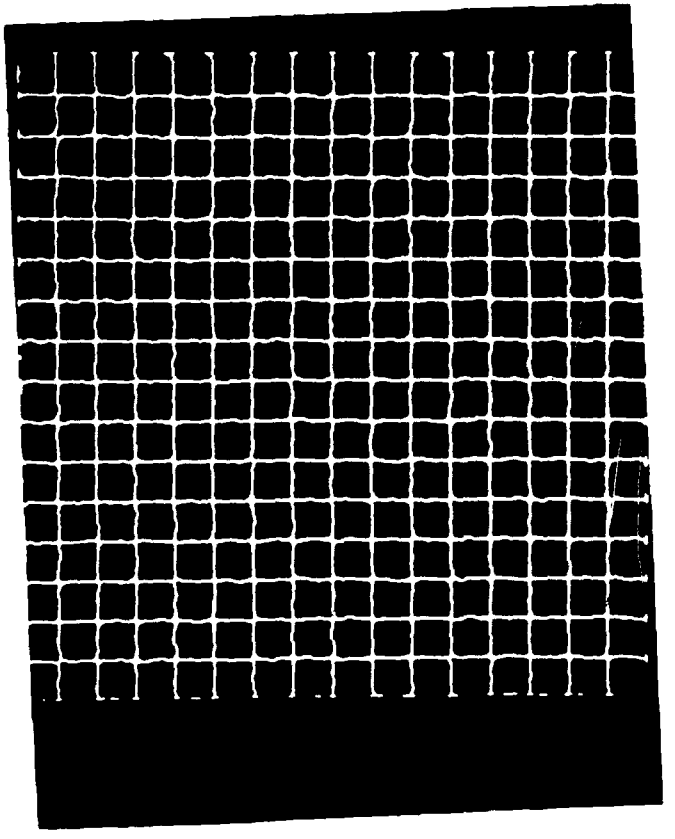


FIGURE 9. Processed edge window ($^{\sigma}N = 50$, $^{\sigma}M = 5$)

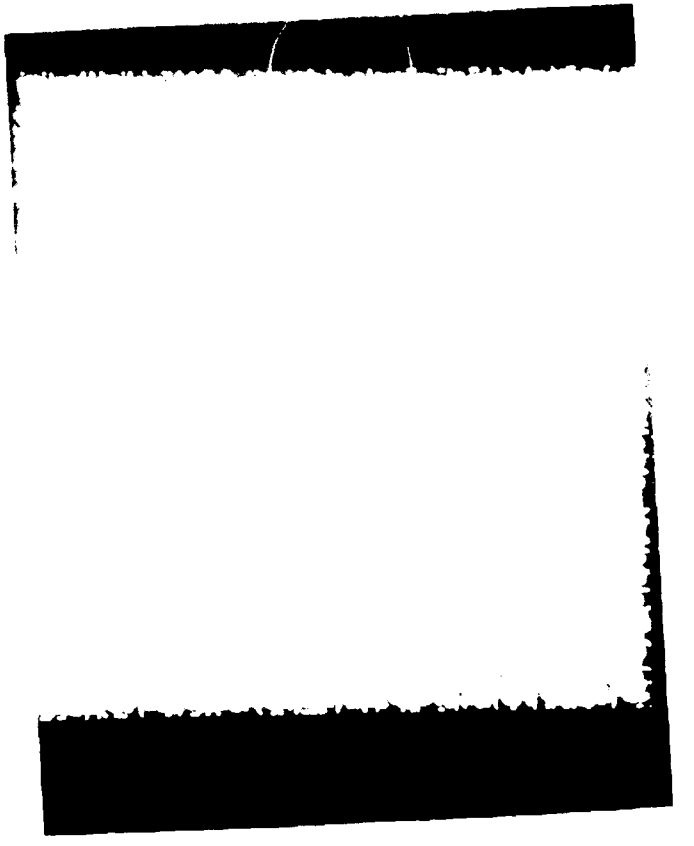


FIGURE 10. Edge window ($^{\sigma}N = 50$) - $^{\sigma}M = 2$

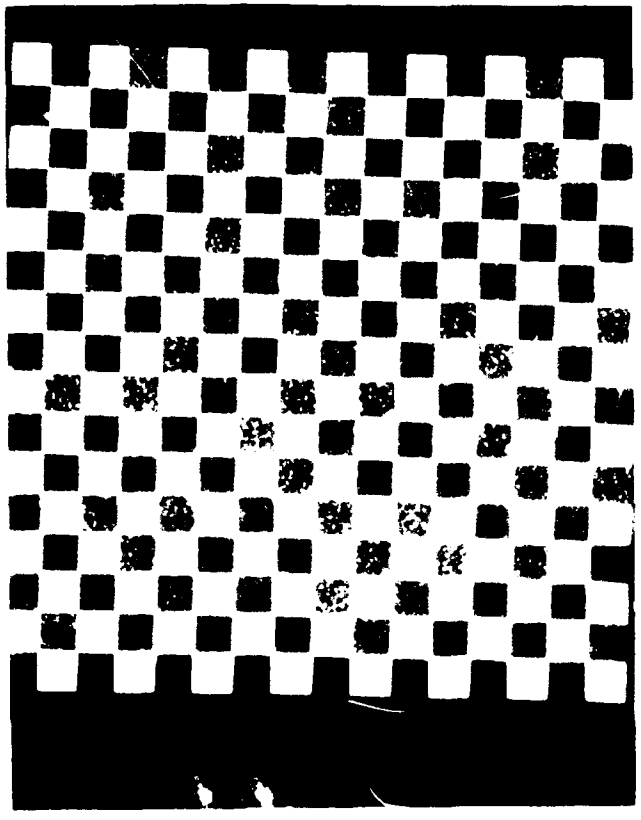


FIGURE 11. Checkerboard #1-Noise Model #2

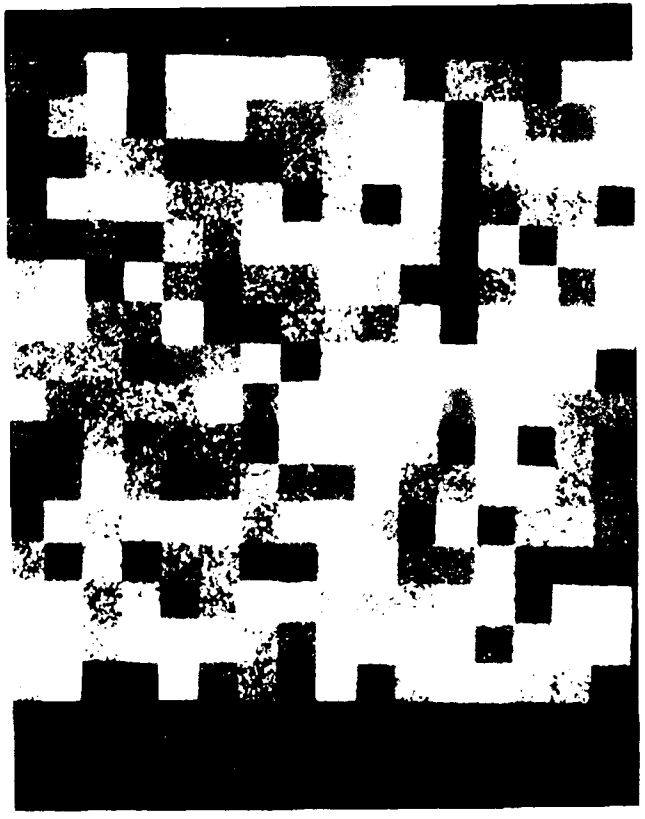
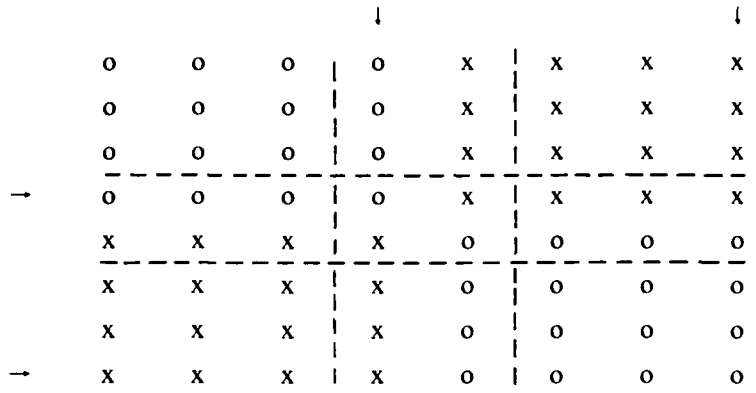


FIGURE 12. Checkerboard #2

FIGURE 13. Checkerboard Edge Definition



o: BLACK SQUARES

x: WHITE SQUARES

Haralick defines an edge pixel as one which has a neighbor that has a different value than the pixel; the two pixel-wide corridors defined by the dashed lines are edge pixels according to Haralick. We defined a pixel to be an edge pixel if its neighbor to the right had a different value, and or if its neighbor just below it had a different value. Thus, the pixel columns and pixel rows indicated by arrows are edges according to our definition. When we processed a noiseless checkerboard with the Marr operator, we found $P_1 = P_2 = 1.000$ and $E = 0.00$, which indicates an unbiased processor.

An examination of tables 1 through 4 indicates that the line-thickening line-thinning process had a slight positive effect on the error E and a slight negative effect on the conditional probability estimate P_2. The operation had a significant positive effect on P_1. That the Marr operator compares favorably with the Prewitt and Haralick operators can be checked by noting that $P_1 = 0.674$ and $P_2 = 0.687$ for the Prewitt operator and that $P_1 = 0.721$ and $P_2 = 0.720$ for the Haralick operator. From table 4, $P_1 = 0.712$ and $P_2 = 0.705$ for the Marr operator when line-thickening, line-thinning was not used. From the same table, $P_1 = 0.814$ and $P_2 = 0.709$ for the Marr operator when line-thickening, line-thinning was used. Haralick determined that $P_1 = 0.721$ and $P_2 = 0.716$ for the Marr operator when he used a 50 by 50 window rather than an 11 by 11 as he did originally. Note that the conditional probabilities in this paragraph pertain to the experiment where $\sigma N = 50$ (Noise) and $\sigma M = 5$ (Marr Resolution).

Consider tables 5, 6 and 7 and especially table 5. The results displayed in these tables demonstrate a marked improvement in the error E when members of the raw primal sketch are combined as described in the section on Numerical Experiments. The improvement in E improves most when pairs from the raw primal sketch set are combined that differ most in their resolution values. The same is true for the conditional probabilities P_1 and P_2. Note that P_2 in the combined form improves over the P_2 components from the raw primal sketches whereas the conditional probability P_1 diminishes somewhat.

The only difference between (Checkerboard #1 - Noise Model #2) and (Checkerboard #1 - Noise Model #1) is that the noise standard deviation varied from square to square in the former and was fixed at five levels (σN = 10, 20, 30, 40, and 50) for the latter. The noise standard deviation varied randomly between σN = 0 and σN = 50 for (Checkerboard #1 -Noise Model #2) as described in appendix B. The results for this model should be similar to the average results of (Checkerboard #1 - Noise Model #1). The validity of this supposition is supported when table 8 is compared to tables 2, 3, and 4.

Checkerboard #2, described in appendix B, should and did cause more problems than any of the other models considered in this work. The gray shade average varied randomly ($75 < GS < 175$) from square to square as did the standard deviation of the noise ($0 < \sigma N < 50$). The resultant noisy image is shown in figure 12 and the processed results are shown in table 9. This model may be more realistic than the other models considered in this report. The likelihood of adjacent squares (fields?) having nearly the same reflectance values is not remote in reality nor in this model. Should both squares have large and nearly equal noise properties, then the boundary might very well be impossible to discern.

The purpose of the work described in this report was to demonstrate that the Marr edge finder compares favorably to the Prewitt and Haralick edge finders especially when simple raster processing functions are applied to the Marr edge picture. The numerical results given in the third paragraph of this section show this, at least as far as noisy checkerboards are concerned. It appears that the performance differences among the several edge finders is trivial when consideration is given to the fact of their inability to find obscured edges. Until considerable research is performed in the development of knowledge bases to help locate hidden edges, we feel that the current crop of edge finders would be best used in an interactive mode where any supporting image processing functions such as those described in appendix C could be used to enhance the sought-for edges.

CONCLUSIONS

1. The Marr edge finder does not appear to be a biased operator.
2. The Marr edge finder compares favorably in performance with the Prewitt and Haralick edge finders.
3. Using noisy checkerboards in an edge detector comparative analysis produces results about noisy checkerboards, a phenomenon few people find useful.
4. Clutter, not random noise, causes most of the problems for edge finders.
5. Future comparative analyses should use cluttered aerial images as bases for tests.
6. Knowledge bases need to be developed to fill in obscured edges.

APPENDIX A

Marr Edge Operator

The Marr edge detector is defined to be the La Placian of the Bivariate Normal density function. Symbolically, the operator is represented as $\Delta^2 G(x,y)$ where Δ^2 is the La Placian and G(x,y) is the normal density function.

$$G(x,y) = (1/2\ \pi\ \sigma^2)\ e - 1/2\ \sigma^2\ (x^2 + y^2)$$

Δ^2: Second derivative operator.

$$\Delta^2 G(x,y) = (2\sigma^2 - R^2)\ e$$

$$R^2 = x^2 + y^2$$

Note that the coefficient $1/(2\pi\sigma^6)$ of $\Delta^2 G(x,y)$ has been discarded and that the function was multiplied by -1. Zero values of $Z = \Delta^2 G(xy)*I(xy)$ are determined where "*" represents a windowing operation and I(x,y) represents the image.

The function G(x,y) is a smoothing function where large values of σ (resolution parameter) cause the operator to act as a low pass filter. Zero values of the second derivative are those places where the gray shade intensity change is greatest.

We approximated the Marr operator with a 49 by 49 window for all of the resolution parameters used in this report. Convolution techniques were used to develop $\bar{Z}$. An edge picture was developed by determining those pixel locations of $\bar{Z}$ where a zero crossing occurred. We did not threshold the zero crossings to determine an edge point. The edge picture was calculated by replacing a pixel with a "1" if a zero crossing occurred, otherwise the pixel was replaced with "0".

APPENDIX B

Checkerboard Models

The basic uncontaminated checkerboard pattern used in this work is larger than the pattern used by Haralick or by Grimson and Hildreth. In their examples, they used a 200 by 200 image where each of the squares of the checkerboard was 20 by 20. We used a 512 by 512 image where each square was 32 by 32. In both cases, ours and theirs, the grayshade value of dark squares was 75 and that of the light squares was 175. We considered two noise models for the basic checkerboard.

Noise Model #1

A random number generator was used to estimate normal deviates $N(o,\sigma N)$ and these estimates were added to each pixel of the uncontaminated checkerboard. We allowed σN to be 10, 20, 30, 40 and 50. Haralick and Grimson and Hildreth considered only $\sigma N = 50$. If the gray shade value plus noise was less than zero, the resultant gray shade was set to zero. If the gray shade value plus noise was greater than 255, the resultant gray shade was set to 255.

Noise Model #2

A random number generator was used to estimate normal deviates $N(o, \sigma IJ)$ and these estimates were added to each pixel of the uncontaminated checkerboard. The standard deviation for the IJth square (I = 1, 16 and J = 1, 16) was itself determined in a random manner.

$$\sigma IJ = 50 * R$$

where R for each IJ was randomly selected from the rectangular distribution $R(\frac{1}{2}, 1/12)$

Neither Haralick nor Grimson and Hildreth considered this model.

Checkerboard Model #2

In this model both the IJth average gray shade value and the IJth standard deviation were selected in a random manner. The characteristics of the IJth square (I = 1, 16 and J = 1, 16) are defined as $N(G_{IJ}, \sigma_{IJ})$.

$$G_{IJ} = 75 + 100 * R_1$$

$$\sigma_{IJ} = 50 * R_2$$

where R_1 and R_2 were randomly selected from the rectangular distribution $R(\frac{1}{2}, 1/12)$.

Neither Haralick nor Grimson and Hildreth considered this model.

APPENDIX C

Image Processing Functions

Three image functions, namely clutter removal, line thickening and line thinning, were used interactively to clean up edge pictures derived from the Marr operator.

Clutter Removal

This function is used to erase blobs or unconnected raster bit patterns of various sizes. In this work, the symbol "CL" defines the dimension of a square window which is centered on every pixel of the edge picture. If no data passes through the window or lies on the window boundary, then all interior data is erased.

Line Thicken

This function is used to connect line breaks and to fill in holes. A 3 by 3 window is placed over each pixel of the edge picture where if the center value is "1" then the eight neighbors are converted to "1's". In this work, the symbol "LIK" defines the number of times the basic line thickener was applied to the edge picture. Note that the three image processing functions are in effect applied in a parallel fashion. This means that the operators are applied to the original edge picture for all pixels and a processed edge picture is produced.

Line Thinner

This function is used to reduce a thickened line of arbitrary pixel width to a center line of no more than one pixel in width. Again a 3 by 3 window is placed over each pixel of the edge image and the algorithm decides whether or not to erase the center pixel. There are 256 cases for the algorithm to consider. In this work the symbol "LTN" defines the number of times the basic line thinner was applied. Note that the process will not disconnect portions of a connected thick line no matter how many times the process is applied. Thus, a disconnected line on the original edge picture may be thickened enough times to fill in the disconnects and then thinned enough times to produce a connected one pixel wide line.

END

12-86

DTIC