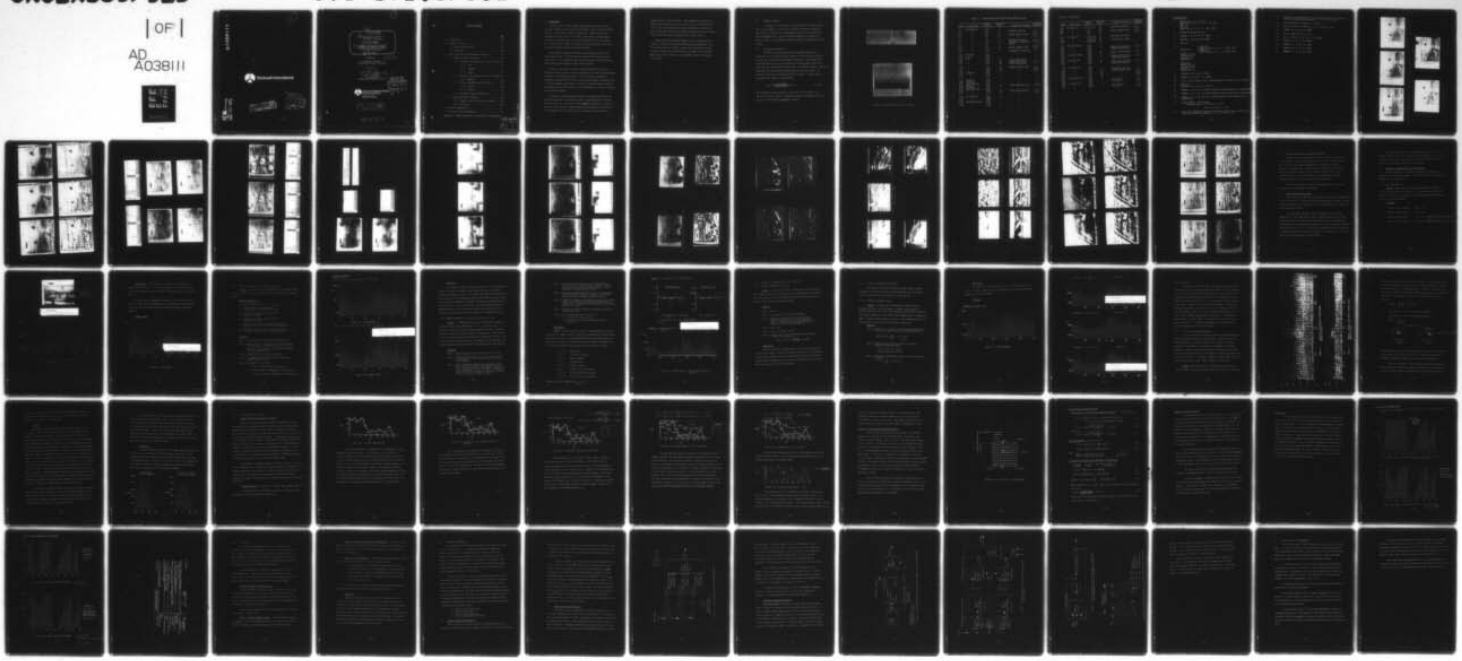
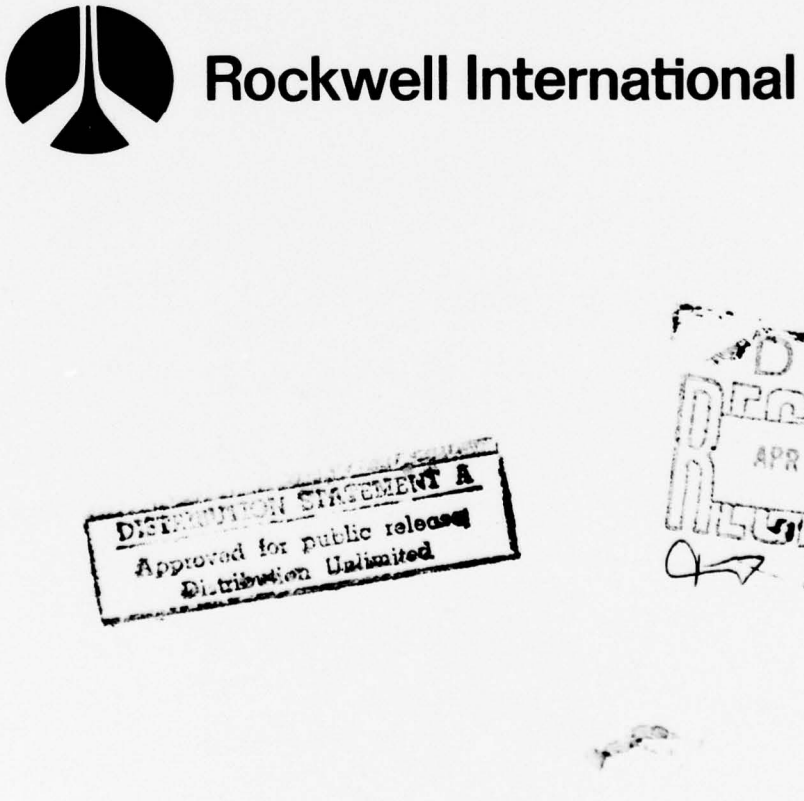
AD-A038 111 ROCKWELL INTERNATIONAL ANAHEIM CALIF ELECTRONICS RES--ETC F/G 17/5
INVESTIGATION OF IMAGE PROCESSING FOR FLIR.(U)
MAY 76 DAAG53-75-C-0245
UNCLASSIFIED C75-671.6/501 NL

| OF |
AD A038111



END

DATE FILMED
4-77



Rockwell International

C75-671.6/501

INVESTIGATION OF IMAGE PROCESSING FOR FLIR.

FINAL REPORT.

CONTRACT NO. DAAG53-75-C-0245
PURCHASE REQUISITION NO. T752587
2141/E294/75 PAN525250

May 1976

67 p.

Prepared by:

Information Sciences
Advanced Technology Department

Approved by:

C. F. O'Donnell
Vice President
Electronics Research Division

Electronics Research Division
Rockwell International
3370 Miraloma Avenue
Anaheim, California 92803

407912

TABLE OF CONTENTS

PAGE

1.0 Introduction 1-0

2.0 Summary of Results 2-0

2.1 Display Characteristics 2-0

2.2 Photographs 2-2

3.0 Image Restoration and Enhancement Techniques 3-0

3.1 Image Restoration Techniques 3-0

3.1.1 Nonlinear, One-Dimensional, Local Techniques 3-1

3.1.1.1 CHOP 3-1

3.1.1.2 DESPIKE1 3-4

3.1.1.3 DEMESA 3-6

3.1.2 Nonlinear, Two-Dimensional, Local Techniques 3-9

3.1.2.1 DESPIKE2 3-9

3.1.3 Linear, One-Dimensional Techniques 3-10

3.1.3.1 LOWPASS 3-10

3.1.3.2 FOURFILT 3-13

3.1.4 Interframe Averaging Experiments 3-13

3.2 Image Enhancement Techniques 3-18

3.2.1 Dynamic Contrast Normalization (IMSHADE) . . 3-18

3.2.2 Contouring 3-31

4.0 Hardware Implementation 4-1

5.0 Conclusions and Recommendations 5-1

Appendix A. Simple Mechanization for Image Point Interpolation

1.0 INTRODUCTION

In FLIR, as in other high dynamic range images, the dynamic range of the scene frequently exceeds that of the display. Scenes containing bright objects, or the horizon, require the operator to readjust the display parameters for each portion of the scene. Under rapidly changing image condition, the operator with all his other duties is not able to respond and is thus forced to leave the settings in a nominal position, thus viewing nearly everything under non-optimum conditions.

Proper image enhancement techniques can intelligently decrease the total dynamic range of the image data by reducing those portions of the signal which do not contribute toward target recognition and identification while preserving and enhancing target related features.

The aim of this effort has been to evaluate Rockwell's previously developed locally developed contrast normalization and other image enhancement techniques to FLIR image processing. The local contrast enhancement technique has demonstrated the ability to match high dynamic range image data to a limited dynamic range display. Additional image enhancement techniques investigated include contouring , one- and two-dimensional frequency filtering, and interframe averaging.

The degree of image enhancement obtained was strongly affected by the signal-to-noise ratio of the available data. Target/background to peak-to-peak noise ratios of less than one were frequently experienced. Linear and nonlinear noise reduction computer programs were generated for the removal of salt and pepper noise, correlated noise, consisting of various types of

ringing as well as fiduciary marks. The programs proved effective in significantly reducing the effects of these noises. Subsequent application of the image enhancement programs demonstrated the ability to improve background contrast ratios without saturating any target characteristics and thus make the background more clearly visible.

Dynamic contrast enhancement appears to be an effective means for freeing the operator from continual display system adjustments. Additional optimization and testing on image data obtained more closely from the focal plane and containing very hot targets and contrasting backgrounds, such as horizons, is recommended to more fully evaluate the technique.

2.0 SUMMARY OF RESULTS

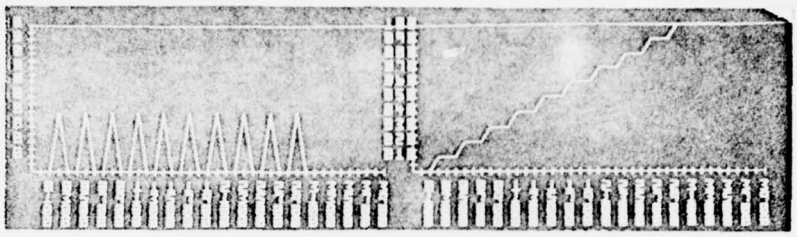
To paraphrase an old saying; "A picture is worth a thousand words and a million numbers." Indeed, the collection of pictures contained in this section is the best way to summarize the results of this FLIR digital image enhancement study. Specific algorithms applied are referenced by entries in Table 2-1 and parameter values are found in the "operation notes" listing.

2.1 Display Characteristics

The CRT/film characteristics and the display software merit discussion, for if uncontrolled their effects on the picture can overwhelm the most skillfully developed enhancement algorithm. A test image and its histogram found in Figure 2-0 illustrate ten uniformly spaced grey levels which span the output system's dynamic range. The display software normalized image point values to produce pictures which utilized this range. The operator provided the values MAXB and MINB of equation 2.1 based upon the histogram of the image to be photographed.

$$I'(i,j) = \left[\frac{I(i,j)-MINB}{MAXB-MINB}\right]\cdot(W-B) + B \qquad \text{Eq. (2.1)}$$

The brightness and bias parameters W and B were set such that B volts provided the control signal to generate the darkest nonsaturated intensity and W volts the brightest nonsaturated intensity.



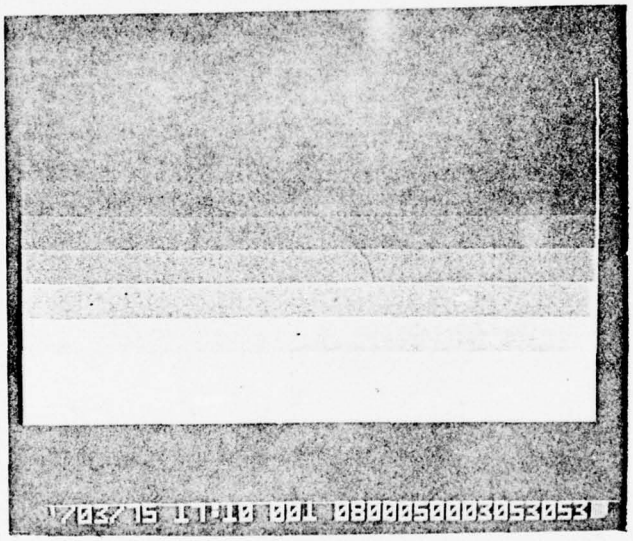


Figure 2-0 Test Pattern and Its Histogram

TABLE 2-1. PHOTOGRAPH DESCRIPTIONS AND CROSS REFERENCE GUIDE

Figure	Description	Figure Processed	Operation Note	Technique Illustrated	Paragraph Reference
2-1	APC/E;Tank/3/4	Orig	-	-	-
2-2	"	2-1	1.	Nonlinear filtering	3.1.1
2-3	"	2-2	2.	"	3.1.1
2-4	"	2-3	3.	Dynamic Contrast Norm.	3.2.1
2-5	"	2-3	4.	"	3.2.1
2-6	"	2-1	5.	Nonlinear filtering	3.1.1
2-7	"	2-6	6.	Nonlinear & lowpass filtering	3.1.3.1
2-8	"	2-6	7.	"	3.1.3.1
2-9	"	2-8	8.	Dynamic Contrast Norm.	3.2.1
2-10	"	2-8	9.	"	3.2.1
2-11	"	2-10	-	Logrithmic Pt. Function	-
2-12	Histogram	2-14	-	-	-
2-13	"	2-15	-	-	-
2-14	APC/E	Orig.	-	-	-
2-15	"	2-14	10.	Coarse Quantization	-
2-16	"	2-14	11.	Nonlinear filtering	3.1.1
2-17	"	2-16	10.	Coarse Quantization	-
2-18	"	2-16	12.	Dynamic Contrast Norm.	3.2.1
2-19	"	2-16	13.	"	3.2.1
2-20	"	2-16	14.	"	3.2.1
2-21	Histogram	2-18	-	-	-
2-22	"	2-19	-	-	-
2-23	"	2-20	-	-	-
2-24	Tank/3/4	Orig.	-	-	-
2-25	Histogram	2-24	-	-	-
2-26	Power Spectrum	2-24	-	-	-
2-27	Tank/3/4	2-24	15.	Linear Bandstop Filter	3.1.3.2
2-28	Histogram	2-27	-	-	-
2-29	Tank/S;2½ Ton/S	Orig.	-	-	
2-30	"	2-29	16	Inter-frame Averaging	3.1.4
2-31	"	2-29	17	"	3.1.4
2-32	2½; Tank; APC	Orig.	-	-	-
2-33	"	Orig.	-	-	-
2-34	"	Orig.	-	-	-
2-35	32x64 Sub-element	2-32	-	-	-
2-36	"	2-33	-	-	-
2-37	"	2-34	-	-	-

Table 2-1. (Continued)

Figure	Description	Figure Processed	Operation Note	Technique Illustrated	Paragraph Reference
2-38	2½; Tank; APC	2-32 to 2-34	18	Inter-frame Averaging	3.1.4
2-39	"	"	19	"	3.1.4
2-40	"	2-32	20	Dynamic Contrast Norm.	3.2.1
2-41	"	2-39	20	"	3.2.1
2-42	2½; Tank; APC	2-32,2-33	21	Difference W/O Offset	-
2-43	"	2-32,2-34	21	"	-
2-44	"	2-32,2-33	22	Difference with Offset	-
2-45	"	2-32,2-34	22	"	-
2-46	Aircraft Carrier	Orig.	-	-	-
2-47	"	2-46	20	Dynamic Contrast Norm.	3.2.1
2-48	"	2-47	-	Logarithmic Function	-
2-49	"	2-46	-	"	-
2-50	"	2-49	20	Dynamic Contrast Norm.	3.2.1
2-51	Plane and Ground	Orig.	-		
2-52	"	2-51	20	Dynamic Contrast Norm.	3.2.1
2-53	"	2-52	-	Logarithmic Function	-
2-54	Plane and Sky	Orig	-		
2-55	"	2-54	20	Dynamic Contrast Norm.	3.2.1
2-56	"	2-55	-	Logarithmic Function	-
2-57	Line Scan IR	Orig.	-	-	-
2-58	"	2-57	-	Logarithmic Function	-
2-59	"	2-57	23	Dynamic Contrast Norm.	3.2.1
2-60	"	2-57	24	"	3.2.1
2-61	"	2-57	25	"	3.2.1
2-62	"	2-57	26	"	3.2.1
2-63	APC/E; Tank/3/4	Orig.	-	-	-
2-64	"	2-63	-	Band Emphasis	3.2.2
2-65	"	2-64	-	Contouring	3.2.2
2-66	2½; Tank; APC	Orig.	-	-	-
2-67	"	2-66	-	Band Emphasis	3.2.2
2-68	"	2-67	-	Contouring	3.2.2

OPERATION NOTES

1. DESPIKE1 (1, 1, 5, 5, 100, 0)
   DEMESA (1000, 100, 5, 1, 1, 1, 1500, 1500)
   CHOP (10)
   CHOP (10)
   DEMESA (500, 0, 75, 1, 1, 1, 4000, 4000)

2. Transpose image and repeat 1.

3. IMSHADE (1, 50, 50, 50, 50, 10000)

4. IMSHADE (1, 50, 50, 80, 80, 10000)

5. CHOP (10)

6. Operation X = { CHOP (10); DEMESA (1000, 1, 5, 1, 1, 1, 1500, 1500); CHOP (10); DEMESA (500,0, 75, 1, 1, 1, 4000, 4000) }
   LOWPASS (50, 10)
   Transpose Image
   Operation X
   LOWPASS (50, 5)

7. Operation X
   LOWPASS (50, 20)
   Transpose Image
   Operation X
   LOWPASS (50, 10)

8. IMSHADE (1, 0, 0, 90, 90, 10000)

9. IMSHADE (1, 20, 20, 75, 75, 10000)

10. A(i,j) = [A(i,j)/1000] · 1000; (IntegerArithmetic and MAX (A(i,j)) $\cong$ 10000)

11. CHOP (10)
    DESPIKE1 (2, 2, 10, 10, 10, 100, 0)

12. IMSHADE (1, 0, 0, 50, 50, 10000) followed by same quantization as operation 10.

13. IMSHADE (1, 0, 0, 75, 75, 10000) followed by same quantization as operation 10.

14. IMSHADE (1, 0, 0, 100, 100, 100, 10000) followed by same quantization as operation 10.

15. Linear Bandstop Filter Program
    Frequencies passed: $0 \leq f \leq 1/32 \quad (\lambda \geq 32)$

    $1/12 \leq f \leq 1/6 \quad (12 \geq \lambda \geq 6)$

16. Image values replaced by mean of clustered data point values from three TV frames using DESPIKE1 algorithm. $n_2 = 1$, $n_3 = 10$, $n_4 = 10$

17. Image values replaced by mean of clustered data point values from five TV frames using DESPIKE1 algorithm. $n_2 = 2$, $n_3 = 10$, $n_4 = 10$.

18. Same as operation 16 except $n_3 = 1$ and $n_4 = 1$.

19. Same as operation 16 except $n_2 = 2$, $n_3 = 1$, and $n_4 = 50$.

20. IMSHADE (1, 0, 0, 99, 99, 10000)

21. $A(i,j) = A_1(i,j) - A_2(i,j) + 5000$

22. $A(i,j) = A_1(i,j) - A_2(i + k_o, j + \ell_o) + 5000$

23. IMSHADE (1, 5, 5, 90, 50, 10000)

24. IMSHADE (1, 0, 0, 90, 90, 10000)

25. IMSHADE (1, 0, 0, 95, 95, 10000)

26 IMSHADE (1, 1, 1, 99, 99, 10000)

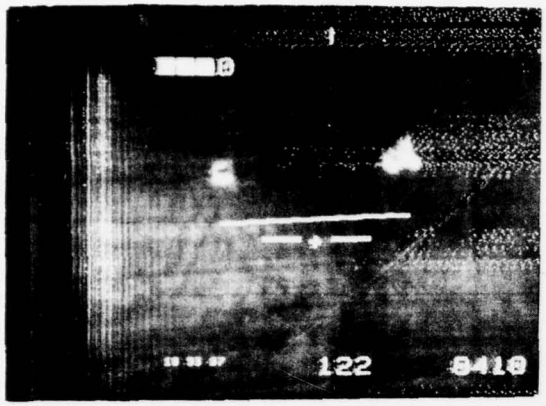


Figure 2-1 APC/E; Tank/3/4 Original

Figure 2-2 Nonlinear, One-Dimensional Filtering (horiz.) of 2-1

Figure 2-3 Nonlinear, One-Dimensional Filtering (vert.) of 2-2

Figure 2-4 Dynamic Contrast Normalization on 2-3

Figure 2-5 Dynamic Contrast Normalization on 2-3

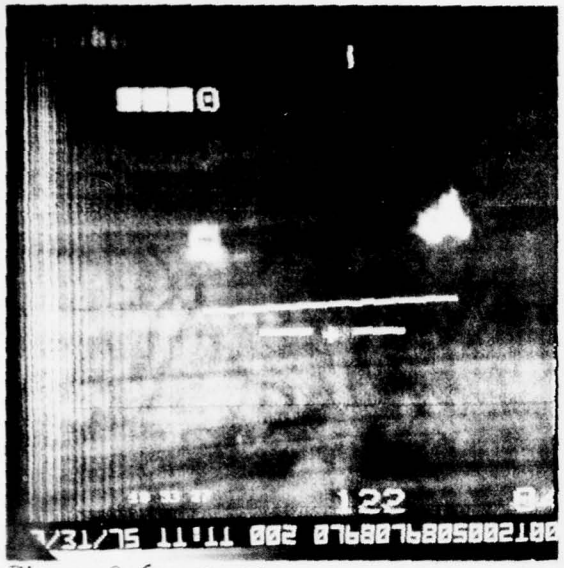


Figure 2-6 Nonlinear, One-Dimensional Filtering of 2-1

Figure 2-7 Linear Lowpass Filtering of 2-6

Figure 2-8 Linear Lowpass Filtering of 2-6

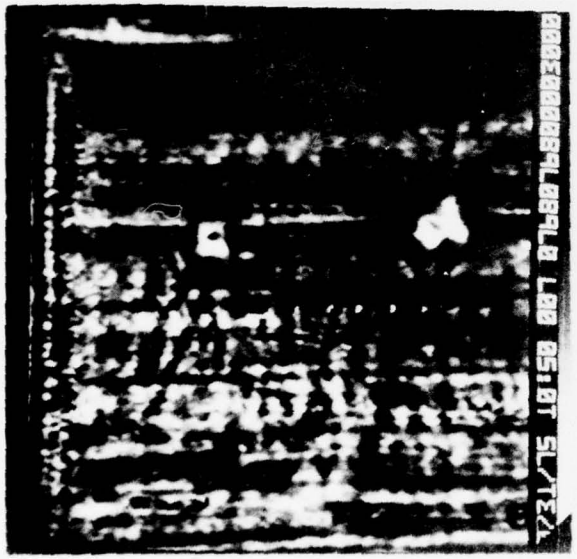


Figure 2-9 Dynamic Contrast Normalization of 2-8

Figure 2-10 Dynamic Contrast Normalization of 2-8

Figure 2-11 Logarithmic Display of 2-10

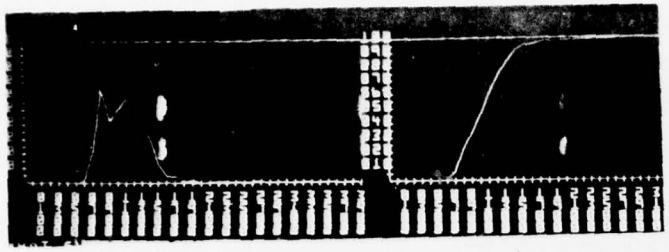


Figure 2-12 Histogram of 2-14

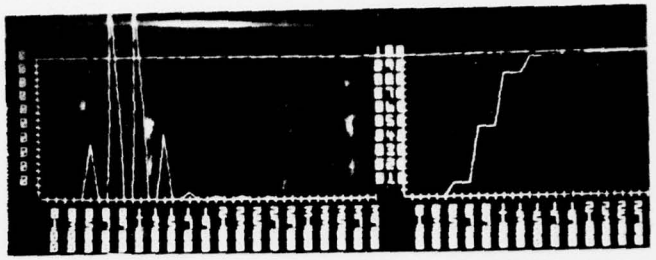


Figure 2-13 Histogram of 2-15

Figure 2-14 APC/E Original

Figure 2-15 Coarsely Quantized Display of 2-14

Figure 2-16 Nonlinear Filtering of 2-14

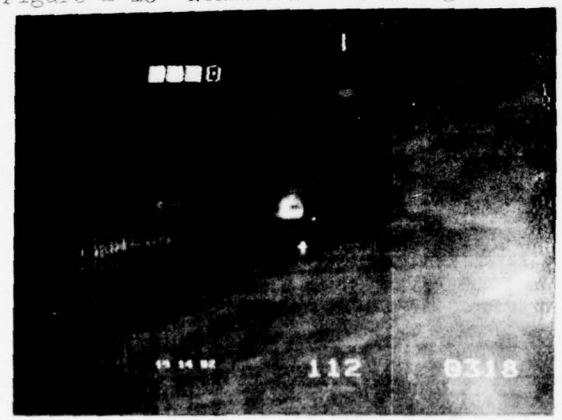


Figure 2-17 Coarsely Quantized Display of 2-16

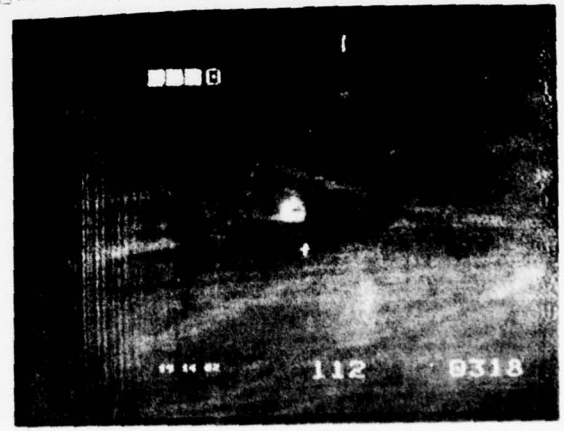


Figure 2-18 Dynamic Contrast Normalization of 2-16

Figure 2-19 Dynamic Contrast Normalization of 2-16

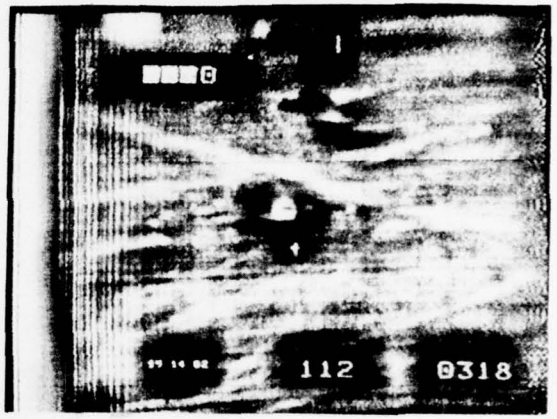


Figure 2-20 Dynamic Contrast Normalization of 2-16

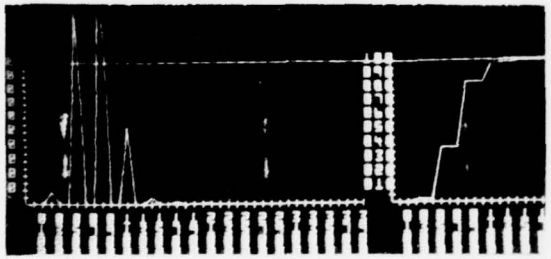


Figure 2-21 Histogram of 2-18

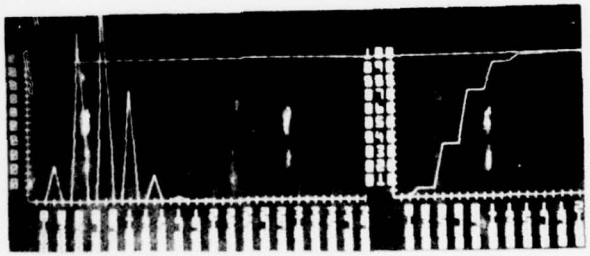


Figure 2-22 Histogram of 2-19

Figure 2-23 Histogram of 2-20

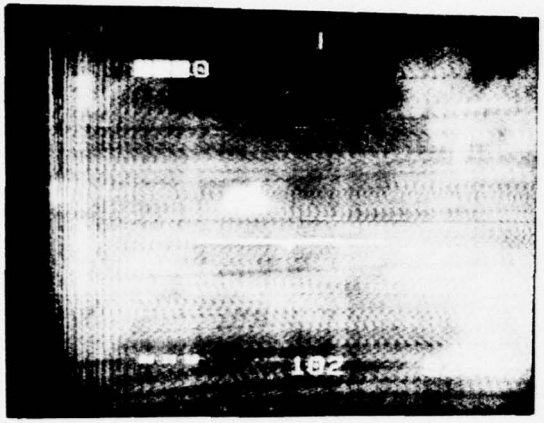


Figure 2-24 Tank/3/4 Original

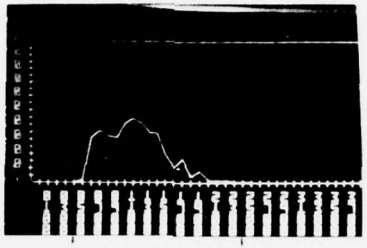


Figure 2-25 Histogram of 2-24

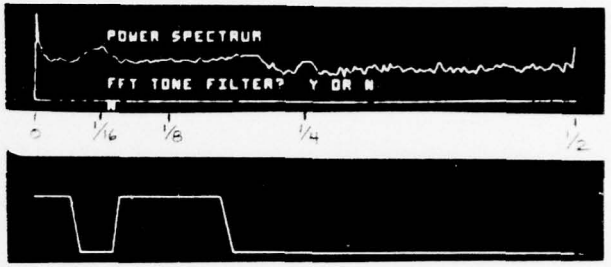


Figure 2-26 Power Spectral Density of 2-24 and Filter Function

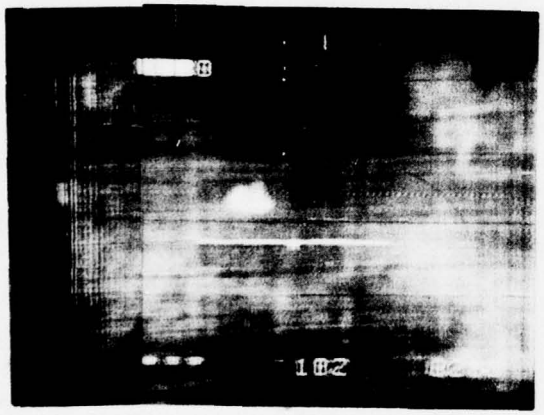


Figure 2-27 Linear Bandstop Filtering of 2-24

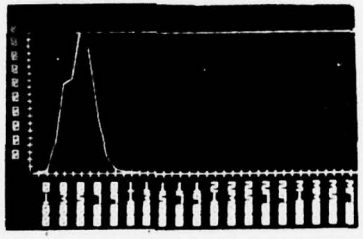


Figure 2-28 Histogram of 2-27

Figure 2-29 Tank/S; $2\frac{1}{2}$ Ton/S Original

Figure 2-30 Interframe Averaging of Scene in 2-29

Figure 2-31 Interframe Averaging of Scene in 2-29

Figure 2-32 $2\frac{1}{2}$ Ton; Tank; APC Original

Figure 2-33 Second View of Scene of 2-32

Figure 2-34 Third View of Scene of 2-32

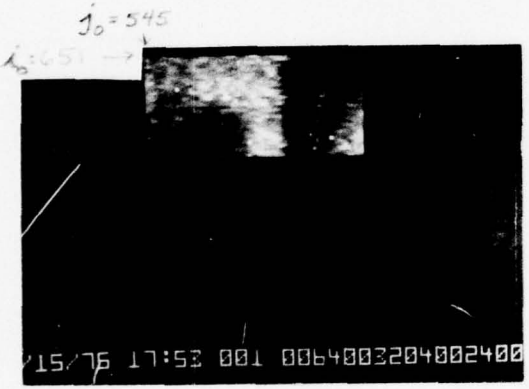


Figure 2-35 Subelement of 2-32

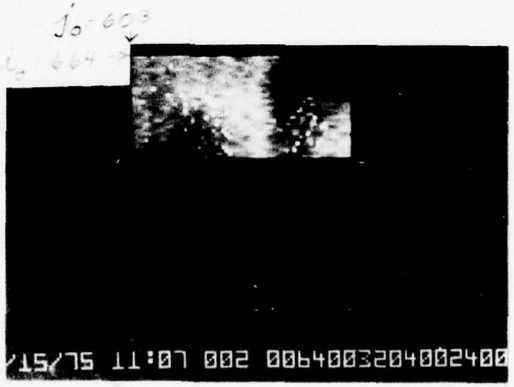


Figure 2-36 Equivalent Subelement of 2-33

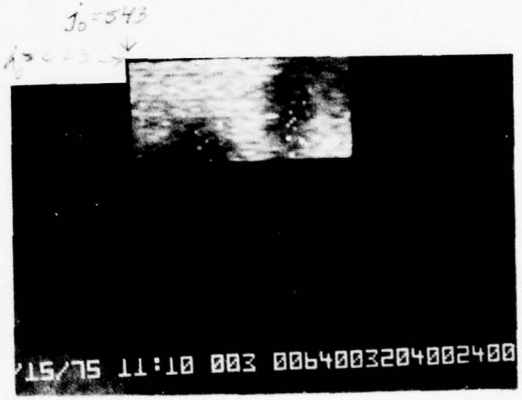


Figure 2-37 Equivalent Subelement of 2-34

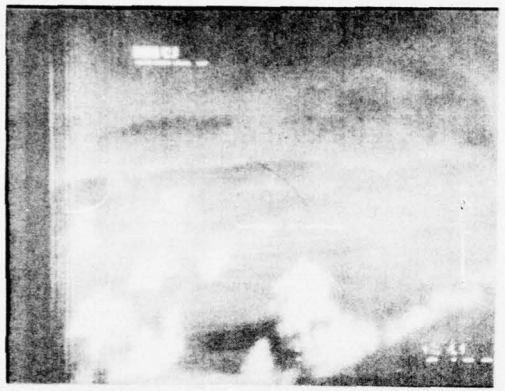


Figure 2-38 Interframe Averaging of 2-32,33,34

Figure 2-39 Interframe Averaging of 2-32,33,34

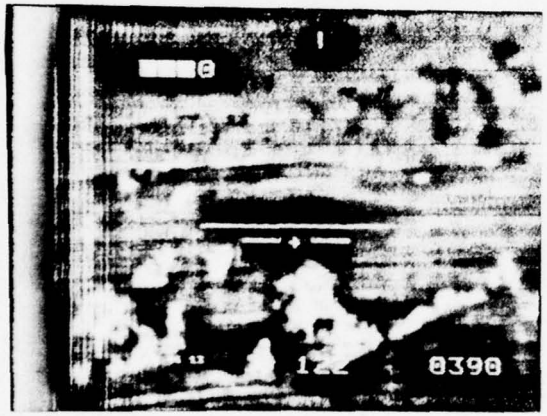


Figure 2-40 Dynamic Contrast Normalization of 2-32

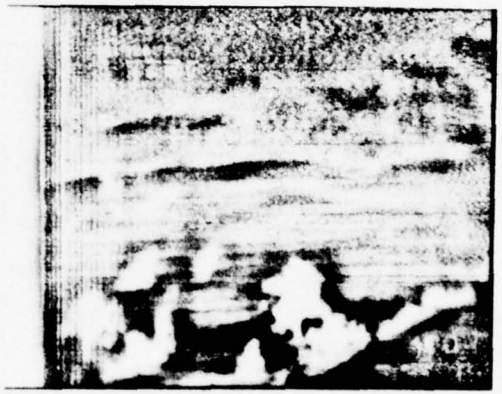


Figure 2-41 Dynamic Contrast Normalization of 2-39

Figure 2-42 Difference Image without Superimposing Subelements

Figure 2-43 Difference Image without Superimposing Subelements

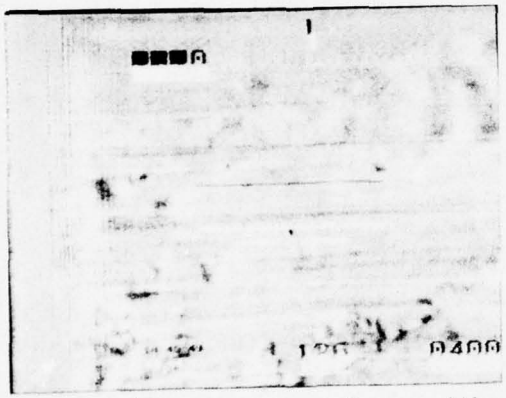


Figure 2-44 Difference Image with Superimposing Subelements

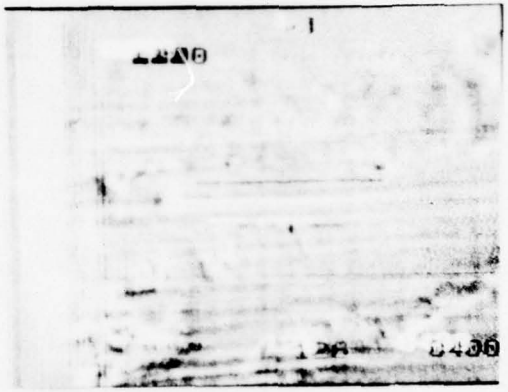


Figure 2-45 Difference Image with Superimposing Subelements

Figure 2-46 Aircraft Carrier - Original (Non-FLIR)

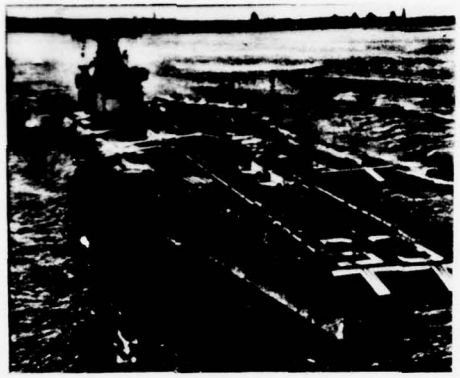


Figure 2-47 Dynamic Contrast Normalization of 2-46

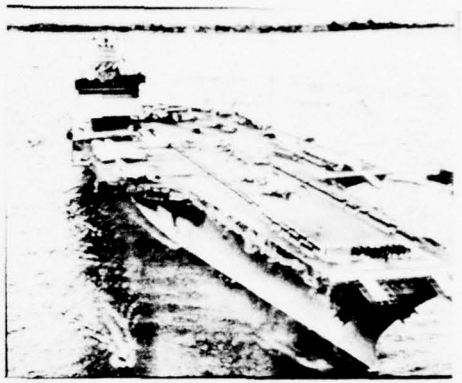


Figure 2-48 Logarithmic Display of 2-47

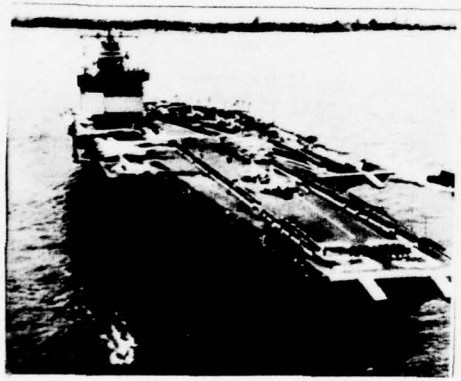


Figure 2-49 Logarithmic Display of 2-46

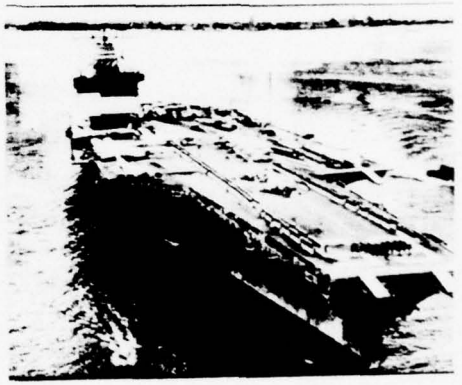


Figure 2-50 Dynamic Contrast Normalization of 2-49

Fig. 2-51 Plane Against Ground - Original (Non-FLIR)

Fig. 2-52 Dynamic Contrast Normalization of 2-51

Fig. 2-53 Logarithmic Display of 2-52

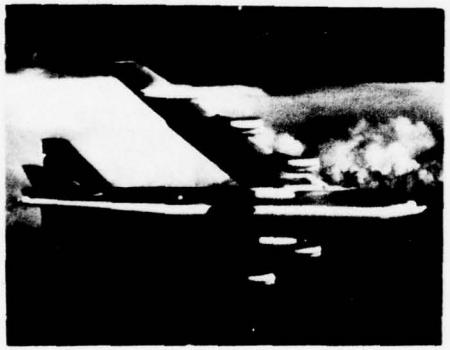


Fig. 2-54 Plane Against Sky - Original (Non-FLIR)

Fig. 2-55 Dynamic Contrast Normalization of 2-54

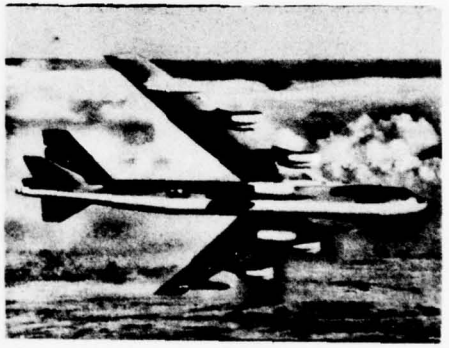


Fig. 2-56 Logarithmic Display of 2-55.

Figure 2-57 Line Scan IR Scene

Figure 2-58 Logarithmic Display of 2-57

Figure 2-59 Dynamic Contrast Normalization of 2-57

Figure 2-60 Dynamic Contrast Normalization of 2-57

Figure 2-61 Dynamic Contrast Normalization of 2-57

Figure 2-62 Dynamic Contrast Normalization of 2-57

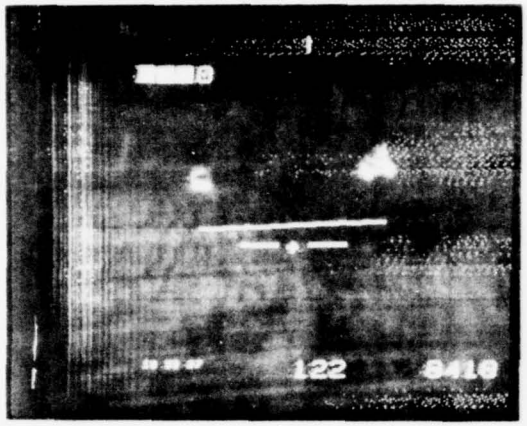


Figure 2-63 APC/E: Tank/3/4 - Original

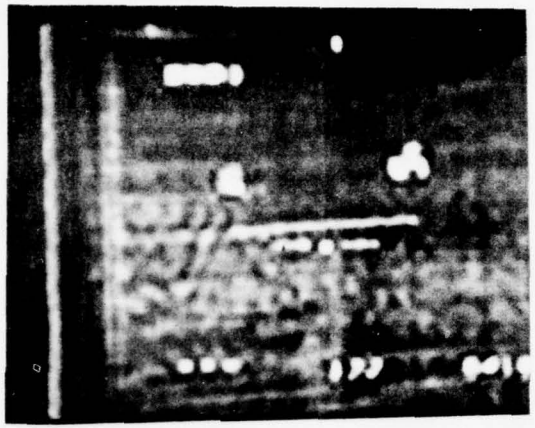


Figure 2-64 Band Emphasis of 2-63

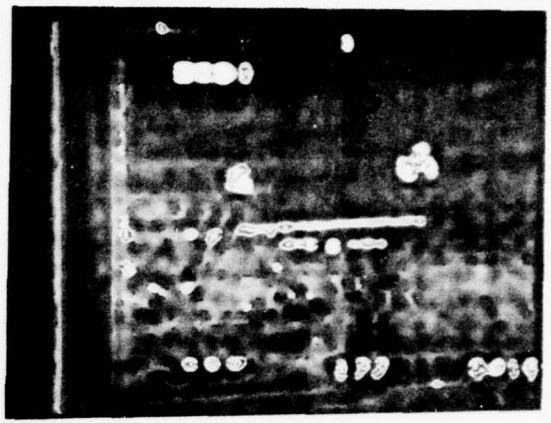


Figure 2-65 Contoured Display of 2-64

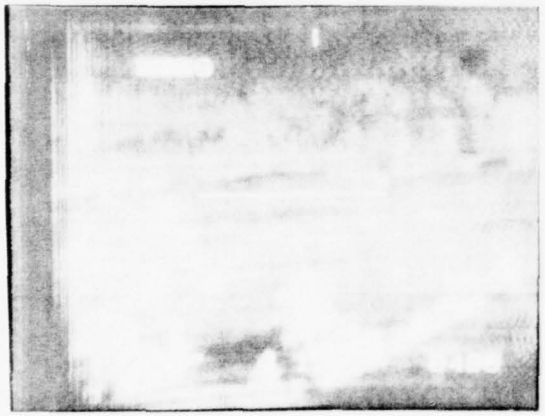


Figure 2-66 2½ Ton: Tank: APC - Original

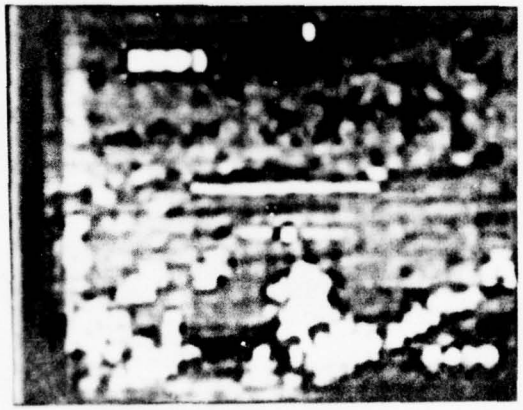


Figure 2-67 Band Emphasis of 2-66

Figure 2-68 Contoured Display of 2-67

3.0 IMAGE RESTORATION AND ENHANCEMENT TECHNIQUES

Image processing algorithms fall into two categories - restoration and enhancement. Restoration algorithms seek to undo the effects of all noise sources in the imaging, digitizing, and transmitting processes. Enhancement algorithms are concerned with the observer and seek to display the scene information optimally for the human's visual system. They deliberately distort the restored image so as to emphasize selectively features most likely to help the observer perform his task.

The image restoration techniques are described first, since they are applied before enhancement is attempted.

3.1 Image Restoration Techniques

Spike noise, data marks, and ringing effects were the main degradations in the FLIR imagery. Nonlinear algorithms were developed to suppress the spike noise and data marks and conventional linear filtering methods were applied to reduce the ringing effects.

One dimensional algorithms operated sequentially on scan lines. Transposition of the image arrays, however, permitted application of these techniques along both vertical and horizontal directions. Two-dimensional algorithms were local operations and required only the eight nearest neighbors' image point values. In all local one and two-dimensional nonlinear algorithms the updated image point values were stored in an output array so as to avoid undesirable and unpredictable cascade effects.

The algorithms' effects are illustrated by cross-section plots of the image intensity functions and power spectral density plots. Figure 3-1 contains the image data which served as the common initial values for the cross-sections and power spectrum plots that follow.

3.1.1 Nonlinear, One-Dimensional, Local Techniques

The CHOP, DESPIKE 1, and DEMESA algorithms were developed to suppress spike noise and data marks. Pictorial results are illustrated in Figures 2-2, 2-3, 2-6, and 2-16.

3.1.1.1 CHOP (Parameter = N)

Concept: Image point values which are extrema with respect to their nearest neighbors are recomputed. The increments are a specified proportion (N/10) of the difference of the original image point value and of one of its nearest neighbors.

Algorithm:

If $A(i-1) \leq A(i) \leq A(i+1)$	$A(i) = A(i)$
If $A(i-1) \geq A(i) \geq A(i+1)$	$A(i) = A(i)$
If $A(i-1) < A(i) > A(i+1)$	$A(i) = A(i) - \frac{N}{10}\left[A(i)-\text{Max}(A(i+1),A(i-1))\right]$
If $A(i-1) > A(i) < A(i+1)$	$A(i) = A(i) - \frac{N}{10}\left[A(i)-\text{Min}(A(i+1),A(i-1))\right]$

Section used for image cross sections and power spectrums

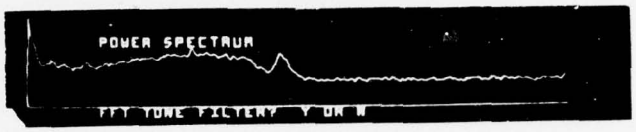


Figure 3-1. Initial Image Data for Figure 3-2 thru 3-8

Application: CHOP's usefulness is limited to the suppression of single pixel spike noise. If its input parameter (N) is set greater than 10, negative image point values can result. Setting it equal to 10 did the best job.

Although only a one-dimensional version of the concept was implemented, it could easily be implemented as a two-dimensional algorithm using the surrounding eight rather than the horizontally-adjacent two nearest neighbors.

Example (N=10)

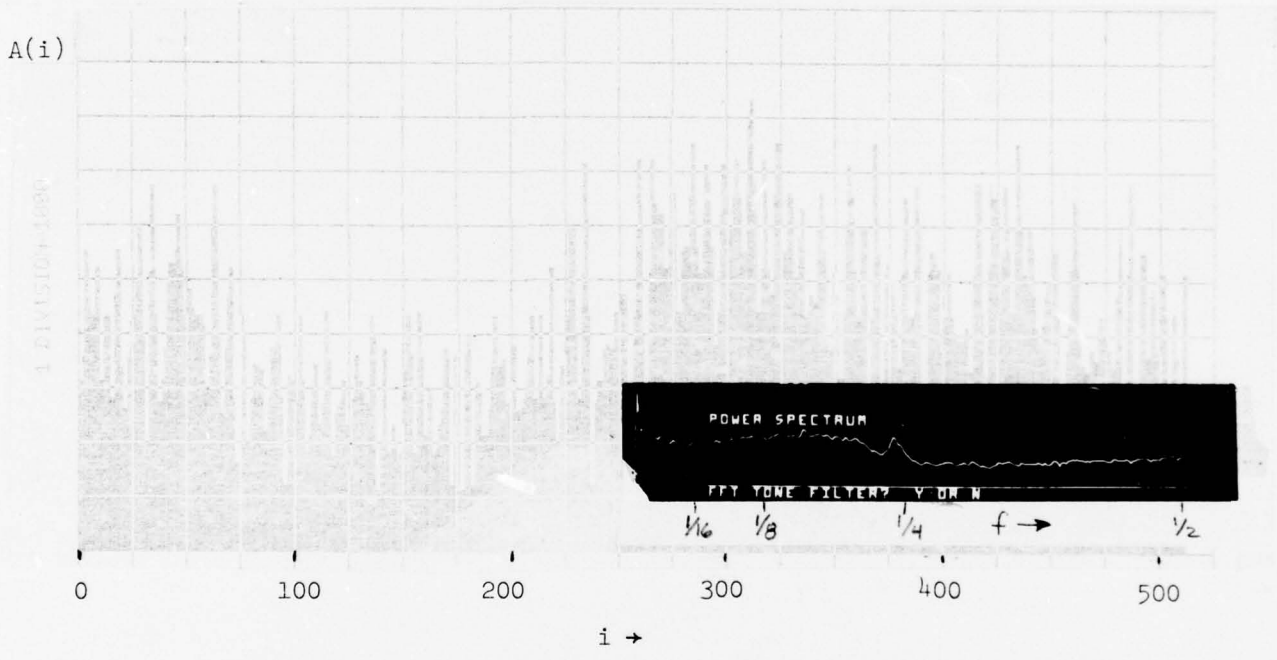


Figure 3-2 CHOP Example

3.1.1.2 DESPIKE1 (Parameters = n_1, n_2, n_3, n_4, n_5, n_6)

Concept: Image point values are replaced with a weighted average of their initial values and the mean values of a clustered set of neighbor points.

Parameter definitions

n_1 - Running mean window = A_{i-n}, ..., A_{i+n_1}

n_2 - Limit to number of points deleted in computation of clustered local mean.

n_3 - Threshold for maximum positive percent deviation from mean.

n_4 - Threshold for maximum negative percent deviation from mean.

n_5 - Weighting coefficient used in update calculation if the original point is excluded from the mean set.

n_6 - Weighting coefficient used in update calculation if original value included in clustered mean set.

Algorithm

Step 1 - Compute mean in running window determined by n_1.

Step 2 - Delete one point (if any) which lies outside limits set by n_3 and n_4. (Measured in units of n_3 and n_4 values). If no points fall outside the range specified, skip ahead to Step 5.

Step 3 - Recompute mean value $\overline{I}(i, n_1, n_2, n_3, n_4)$.

Step 4 - If number of points deleted does not exceed n_2, repeat steps 1, 2, and 3.

Step 5 - Recompute I(i) as follows:

$$I'(i) = (1-\alpha)I(i) + \alpha \cdot \overline{I}(i, n_1, n_2, n_3, n_4)$$

If I(i) deleted in computation of $\overline{I}$, then $\alpha = n_5$.

If I(i) not deleted in computation of $\overline{I}$, then $\alpha = n_6$.

EXAMPLE OF DESPIKE:

Case 1 $n_1 = 2$ $n_2 = 2$ $n_3 = 1$ $n_4 = 1$ $n_5 = 1$ $n_6 = 1$

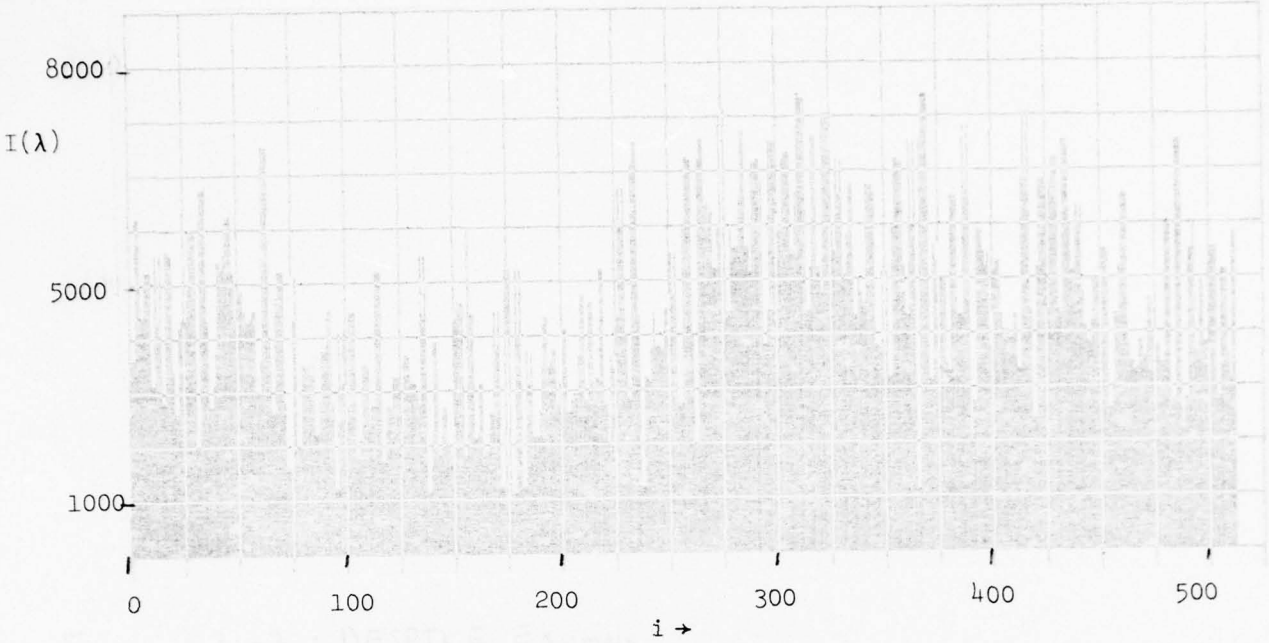


Figure 3-3 DESPIKE Example 1

Case 2 $n_1 = 2$, $n_2 = 3$ $n_3 = 1$ $n_4 = 1$ $n_5 = .75$ $n_6 = .25$

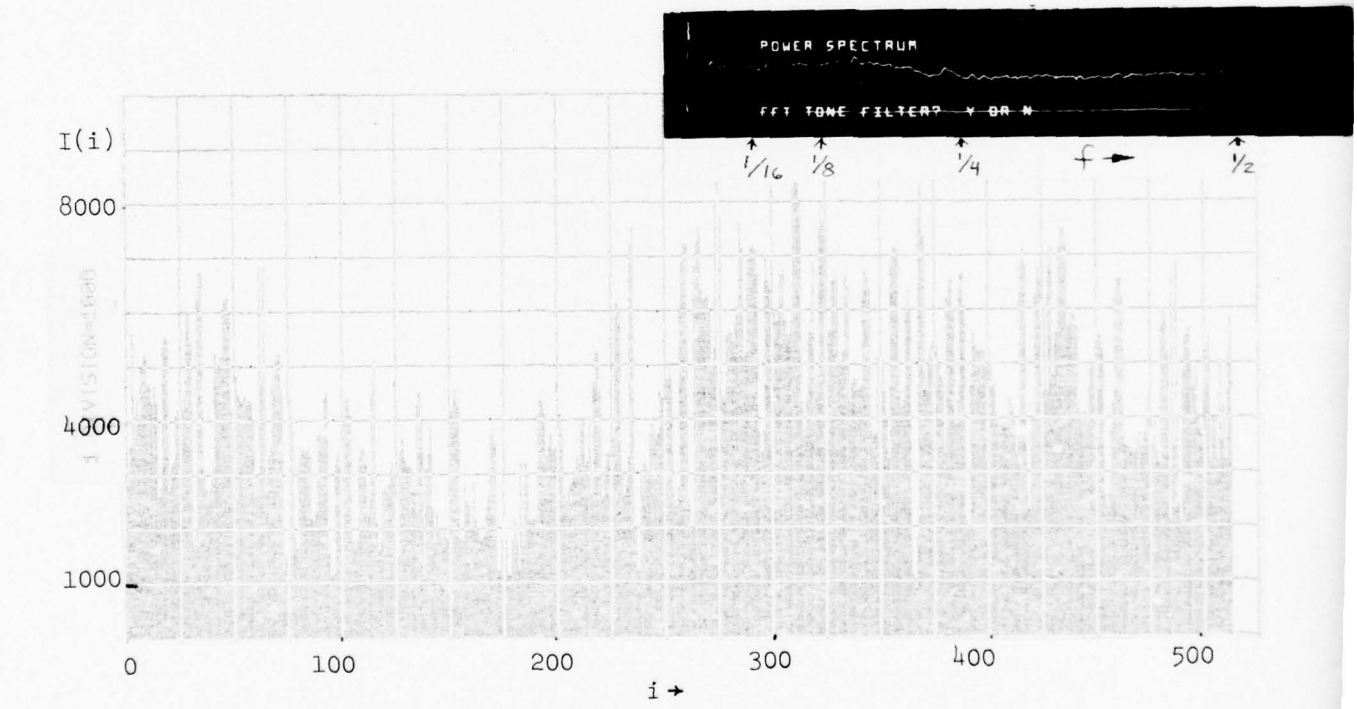


Figure 3-4 DESPIKE Example 2

Application

The DESPIKE algorithm is the most versatile of the one-dimensional nonlinear algorithms developed. Particularly useful are the separate threshold percentages (n_3 and n_4) for positive and negative deviations from the cluster mean and the separate weighting coefficients (n_5 and n_6) which allow completely independent updating of an image point depending upon its deviation from the local clustered mean. The two thresholds provide independent sensitivity to positive and negative spike noise and the two coefficients provide a more selective update of the image point values.

3.1.1.3 DEMESA (Parameters = n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8)

Concept: DEMESA operates on the one-dimensional derivative obtained by taking first differences: $I'(i,j) = I(i+1,j)-I(i,j)$. It searches for paired positive and negative peaks which exceed a specified threshold and which lie within a specified neighborhood. The derivative for all points in these paired peak regions is replaced by the mean of the original derivative function values. The objective of the algorithm is to eliminate spike noise and features with sharp edges (e.g., data markings).

Algorithm

Step 1 - Replace image intensity function by its derivative.

Step 2 - Perform a forward search for a point whose absolute value exceeds n_1. Go to Step 7 if end of scan line is encountered.

Step 3 - Search forward and backward from the point found in Step 2 until the function either changes polarity or until its absolute value is less than n_2. Compute the sum of the values of the set of points within the region defined by this search. If the sum's absolute value is less than n_7, return to Step 2.

Step 4 - Perform another forward search for a distance not to exceed n_3 until a functional value is found of opposite polarity and whose absolute value exceeds n_1. If no such point is found, return to Step 2.

Step 5 - Repeat the operation described in Step 3 in the neighborhood of the peak of opposite polarity. If the sum's absolute value is less than n_8, return to Step 4.

Step 6 - Compute the average of the functional values of all points contained in the two regions defined by the searches of Steps 3 and 5. Replace each of the values by this average and return to Step 2.

Step 7 - Integrate the updated derivative function and proceed to the next scan line.

Options: (1) To skip Steps 1 and 7, set $n_4 = 0$

(2) To iterate N times, set $n_5 = N$

(3) To search only for positive peaks in Step 2, set $n_6 = 1$.

Application

DEMESA's use was limited to removal of data markings which interferred with the dynamic contrast normalization enhancement algorithm IMSHADE. Cross-sectional plots of the image intensity function through targets and data marks were studied to determine the values of DEMESA's parameters that would selectively remove the data marks without affecting the target.

$n_1 = 500$ (threshold)

$n_2 = 1$ (secondary threshold)

$n_3 = 10$ (search distance)

$n_4 = 1$ (differentiate)

$n_5 = 1$ (number of passes)

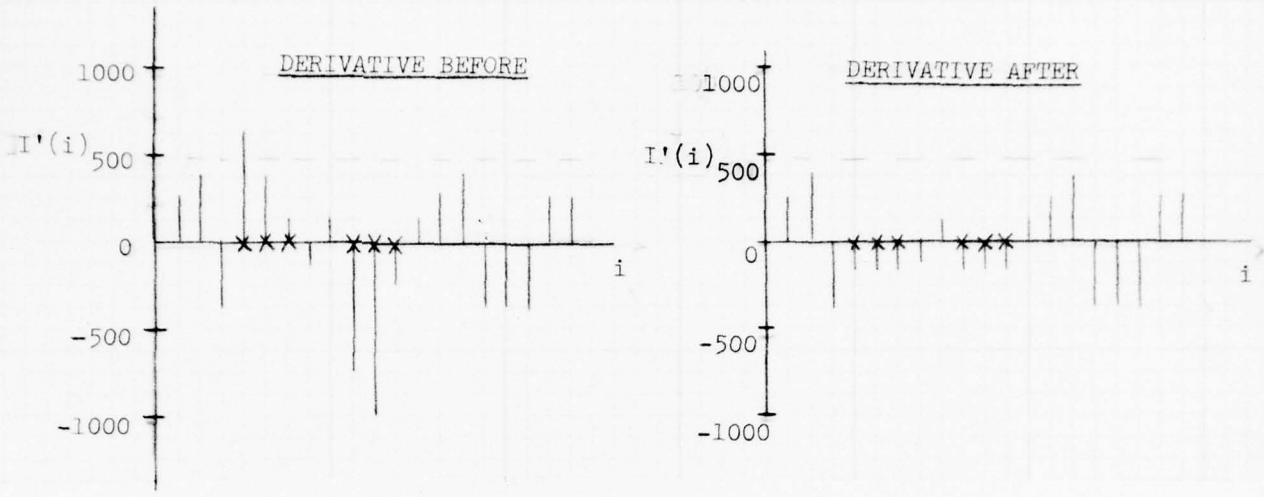
$n_6 = 1$ (trigger on positive spikes)

$n_7 = 500$ (area threshold, first peak)

$n_8 = 500$ (area threshold, second peak)

Parameter values for DEMFSA Examples 1 and 2.

Example 1: Illustrates Cancellation Effects



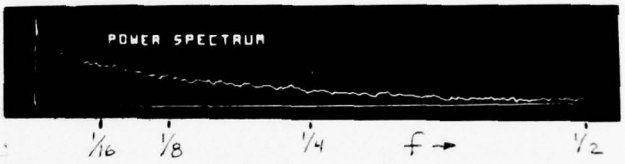
x ⇒ contributes to arithmetic mean.

x ⇒ I(i)=M= -150

M = (700 + 300 + 100 - 750 - 1000 - 250)/6

M = -150

EXAMPLE 2: (Same parameters as Example 1)



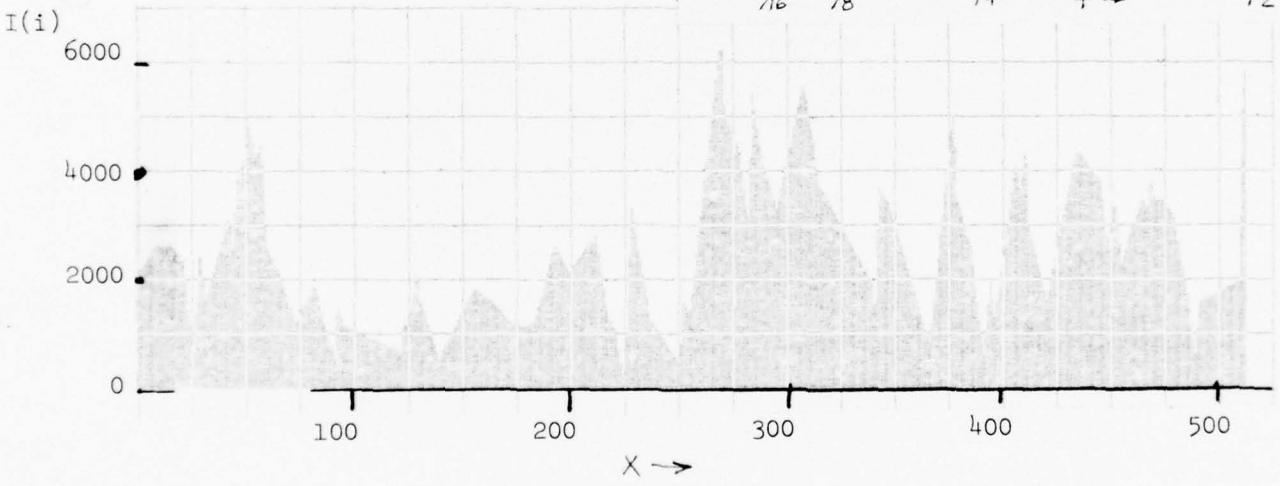


Figure 3-5 DEMESA Example 2 (Same Parameter Values as Example 1)

3.1.2 Nonlinear, Two-Dimensional, Local Techniques

3.1.2.1 DESPIKE2 (Parameters = n_1, n_2)

Concept: The image point value is replaced by a weighted average of its initial value and the mean of 8-N of its surrounding eight neighbors. The N values excluded from the neighborhood mean are those which are rejected in the following clustering algorithm.

Algorithm

Step 1 - Set m = 0

Step 2 - Form average $\bar{M}$ of 8-m nearest neighbors.

Step 3 - Compare absolute value of the 8-m differences between the average of Step 2 and image point values. Eliminate from the nearest neighbor set that point with the largest deviation.

Step 4 - Set m = m+1

Step 5 - If $n_1 < m$, return to Step 2

Step 6 - Compute a new value for the image point according to the following expression

$$I'(i,j) = I(i,j) + \frac{n_2}{100} \cdot \left[\bar{M} - I(i,j) \right]$$

Application

DESPIKE 2 was used to suppress single pixel and single scan line noise from the images. The specific algorithm described is representative of many nearest neighbor nonlinear filter functions which have straightforward hardware implementations. They defy conventional analysis and are based on intuition and heuristic reasoning.

3.1.3 Linear, One-Dimensional Techniques

The LOWPASS and FOURFILT filtering programs removed frequency components in the image function in which noise power was significant. Pictorial results are found in Figures 2-7, 2-8, and 2-27.

3.1.3.1 LOWPASS (Parameters = L, λ_c)

Concept: A conventional one-dimensional lowpass filtering approach was tried on the FLIR images to suppress high frequencies. The high frequency content of the digitized images must be due to additive noise sources because the video disk's 6 MHz bandwidth does not support the number of samples taken along each scan line. This oversampling effect is seen in plots of one scan line's 1024 sampled image points.

Algorithm

Step 1 - Compute $2L + 1$ terms of the inverse Fourier transform of an ideal lowpass filter with cutoff frequency $1/\lambda_c$.

$$h(i) = \frac{i}{\pi} \sin \cdot \frac{2\pi}{\lambda_c \cdot i} \quad \text{for } -L \le i \le L$$

$$L \le 50$$

Step 2 - Extend the image function beyond its borders by generating its mirror image.

$$I(-i) = I(i+1) \text{ for } 0 \le i \le L$$

$$I(N+1+i) = I(N-i) \text{ for } 0 \le i \le L$$

Step 3 - Obtain the filtered image function from the convolution sum below.

$$I'(i) = \sum_{j=-L}^{L} I(i-j) \cdot h(j)$$

Applications

The bandwidth of the noise proved to be so broad that significant amounts of ringing remained at cutoff frequencies for which both target and background details were blurred.

Examples:

FILTERED: L = 50; λ_c = 4:

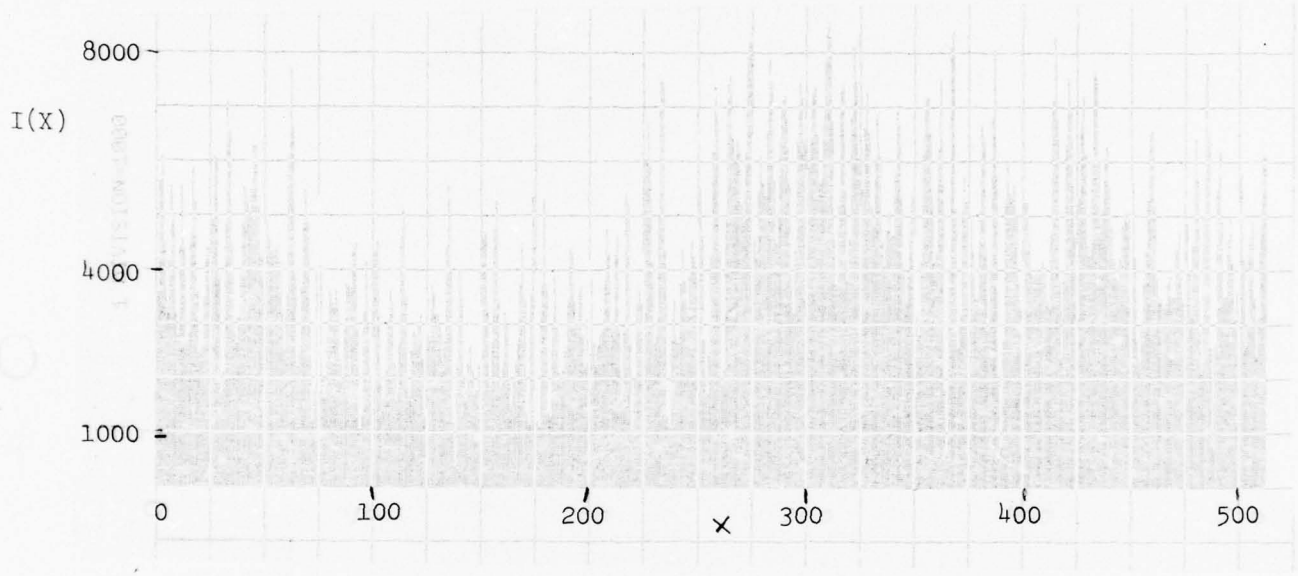
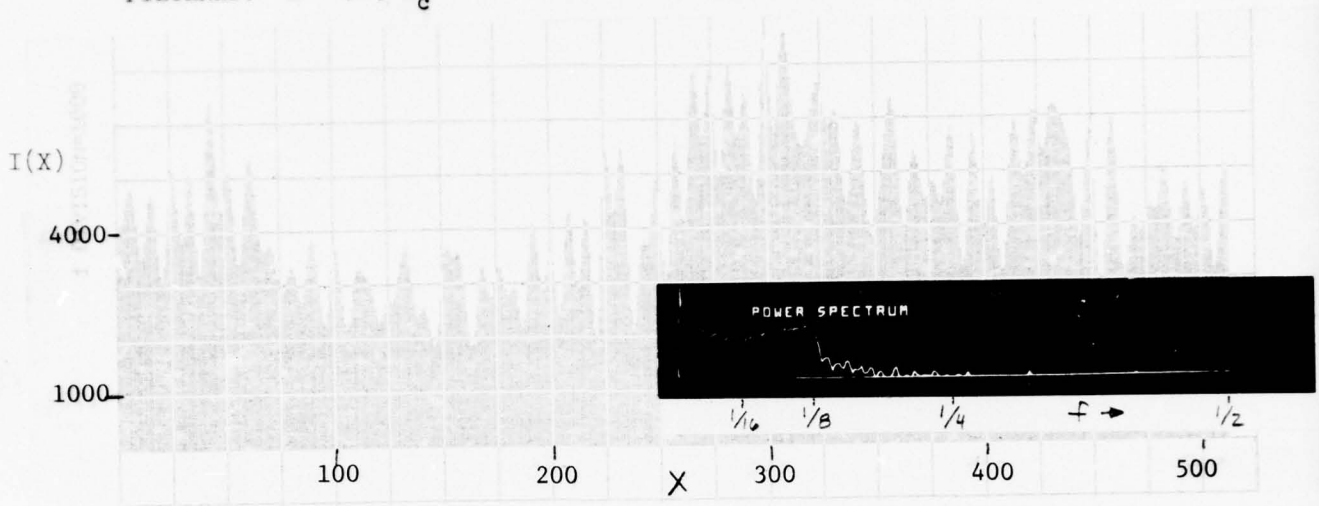


Figure 3-6 Lowpass Example 1

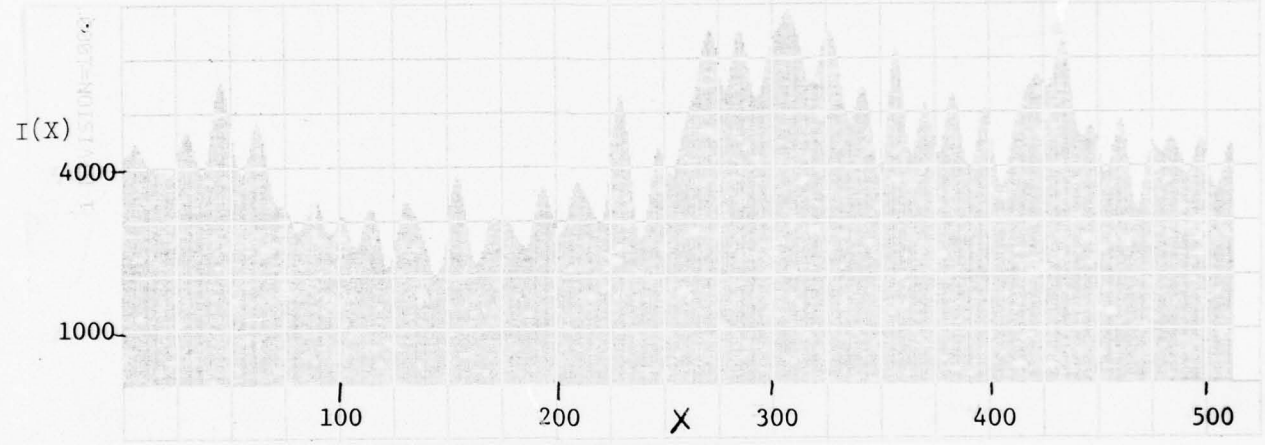
FILTERED: L = 50; λ_c = 8:

Figure 3-7



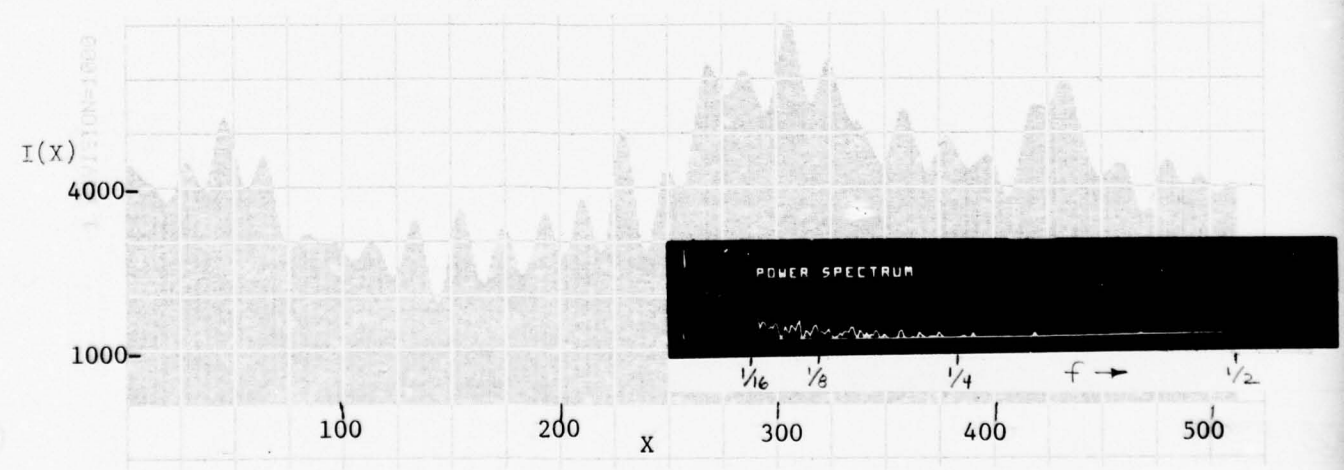
FILTERED: L = 50; λ_c = 12:

Figure 3-8



FILTERED: L = 50; λ_c = 16:

Figure 3-9



3.1.3.2 FOURFILT

The failure of LOWPASS to remove the ringing noise without unacceptable blurring of target and background objects led to consideration of a conventional linear filter with stop bands in those regions of the frequency domain that were dominated by noise. Determination of those regions was based upon inspection of the power spectral density function of the image. Sequential scan lines were concatenated to compute the average power density in a horizontal slice of the image. A stop band was placed where a peak in the function suggested the presence of noise power. High frequency components were also removed. Cross-section plots through the target regions of Figures 2-24 and 2-27 are found in Figure 3-10 and 3-11. A dramatic improvement in the signal-to-noise ratio was achieved by this filtering method.

3.1.4 *Interframe Averaging Experiments*

All of the restoration techniques which have been described thus-far perform spatial filtering on single frames. Another approach that takes advantage of the large difference in the temporal correlation properties of the signal's scene and noise components is interframe averaging. Since the scene intensity varies in time more slowly than much of the noise, performing low pass temporal filtering can improve the signal-to-noise ratio. A consecutive sequence of images taken by a FLIR system was available to test this approach to restoration.

Method: The first step was to determine the horizontal and vertical shifts of the target between successive TV frames. A 32x64 sub-

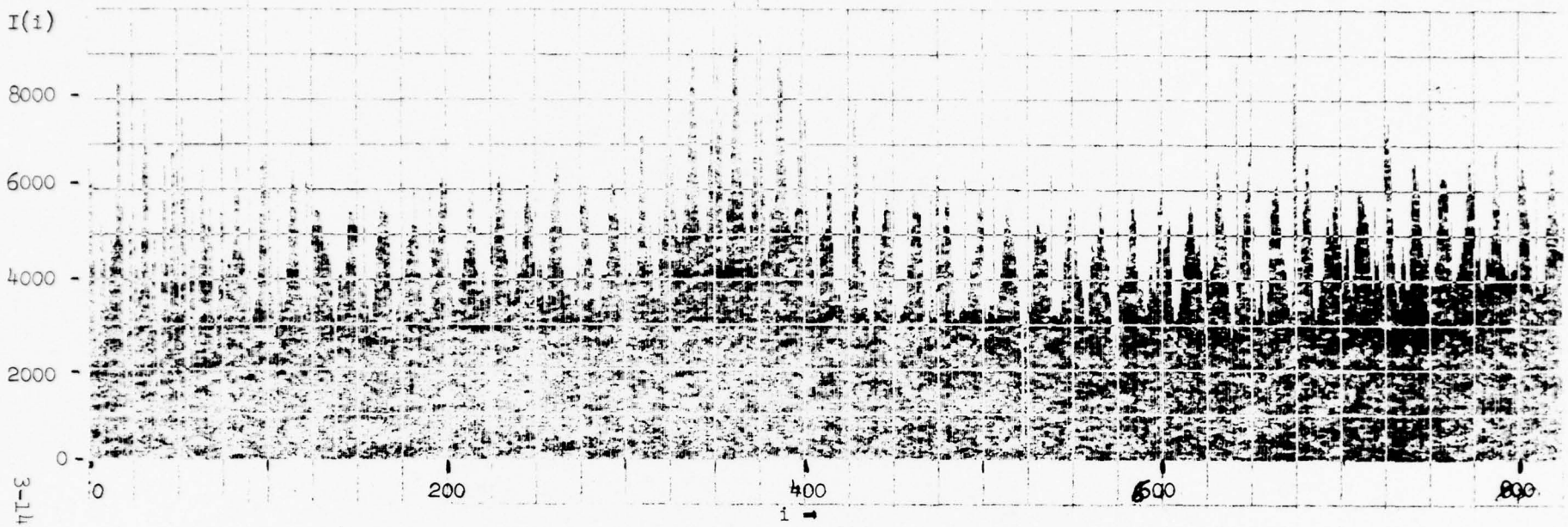


Figure 3-10. Prefiltered, cross-section through target (Refer to Figure 2-24)

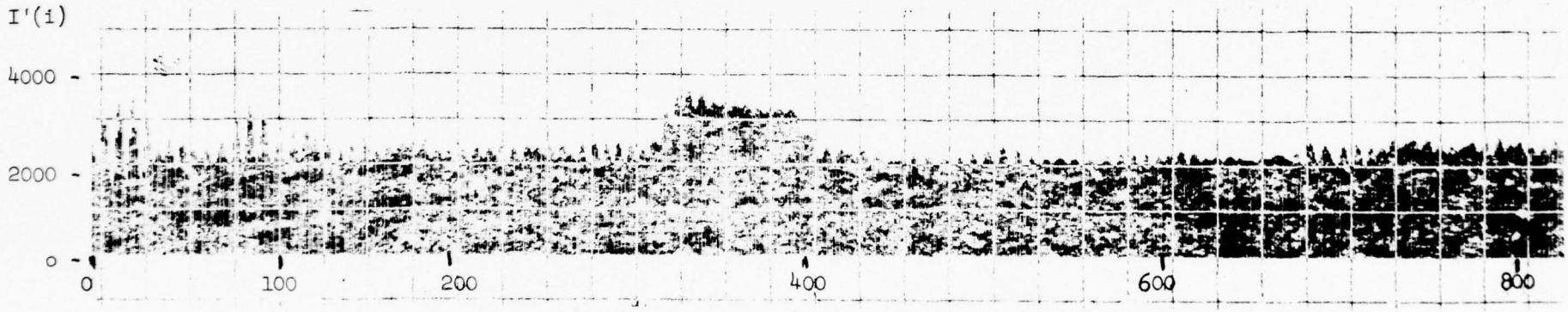


Figure 3-11. Bandstop filtered, cross-section through target (Refer to Figure 2-27)

element containing the target in the first frame of the TV sequence served as a matching template. A simple metric was computed between the template and 32x64 subelements in subsequent frames. A matrix of these scores was generated by displacing the template within a region ±14 pixels horizontally and vertically from nominal. The metric is defined by the following equations:

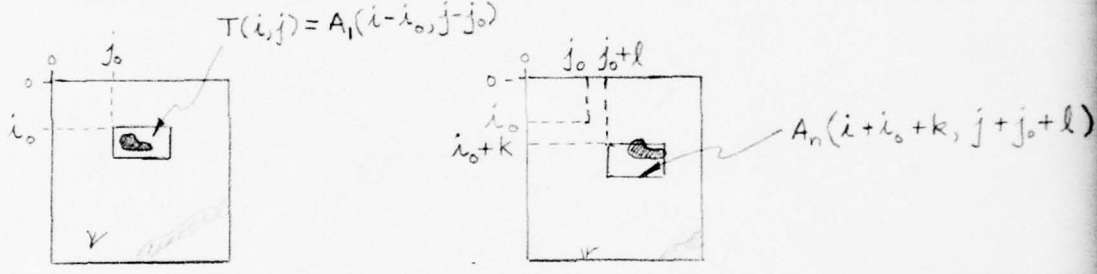
$$M_n(k,\ell) = \sum_{i=1}^{32} \sum_{j=1}^{64} m_n(i,j,k,\ell)$$

$$m_n(i,j,k,\ell) = 1 \text{ if } |T(i,j) - A_n(i+i_o+k, j+j_o+\ell)| \leq t$$

$$m_n(i,j,k,\ell) = 0 \text{ if } |T(i\ j) - A_n(i+i_o+k, j+j_o+\ell)| > t$$

where $-14 \leq k \leq 14$

$-14 \leq \ell \leq 14$



Frame 1 Frame n

The desired horizontal and vertical shifts were obtained from the indicies of the maximum value within the M array. No rotations, scale changes, or perspective changes were taken into account in these experiments.

The second step was to create a composite image by superimposing several images and computing new image point values. Were a Gaussian white noise model assumed, simple averaging would be the optimum calculation to

perform. The spike noise observed failed to fit this model, however, and a DESPIKE1 based algorithm (see para. 3.1.1.2) was applied to compute an average of a clustered set of image points.

Results:

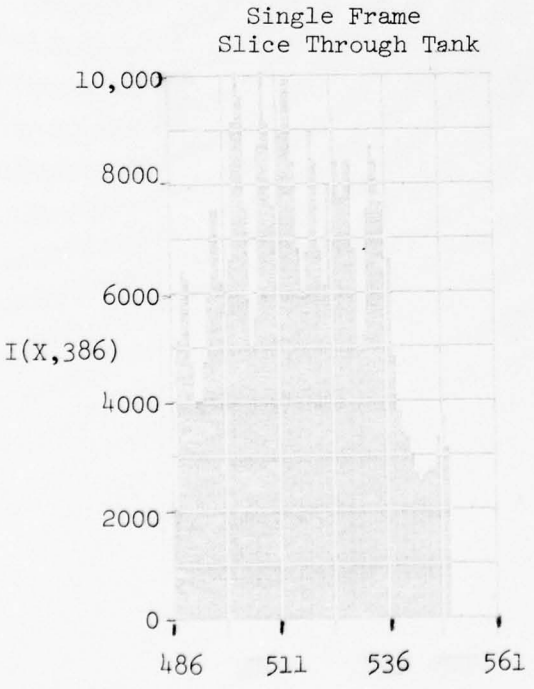
Interframe averaging experiments were performed on two sets of FLIR images. The first set consisted of five consecutive frames of a TV sequence. The first frame and two composite pictures of the scene are found in Figures 2-29 to 2-31. The first composite (Fig. 2-30) was obtained from a point-by-point average of either two or three image points from three consecutive frames. All three points were averaged if no one point deviated by more than 10 percent from the mean. Otherwise, the two values nearest the mean were averaged to compute the composite image's value. The second composite (Fig. 2-31) was obtained from a similar calculation on five consecutive frames. The only difference was that as many as two values could be rejected during the averaging procedure if their values deviated by more than 10 percent from the mean. Images generated by taking pointwise differences (and adding a constant to avoid negative functional values) are displayed in Figures 2-42 to 2-45. The top two were formed without the spatial offset required for superimposing the 32x64 subelements already described. The bottom two were formed from differences of appropriately offset image functions.

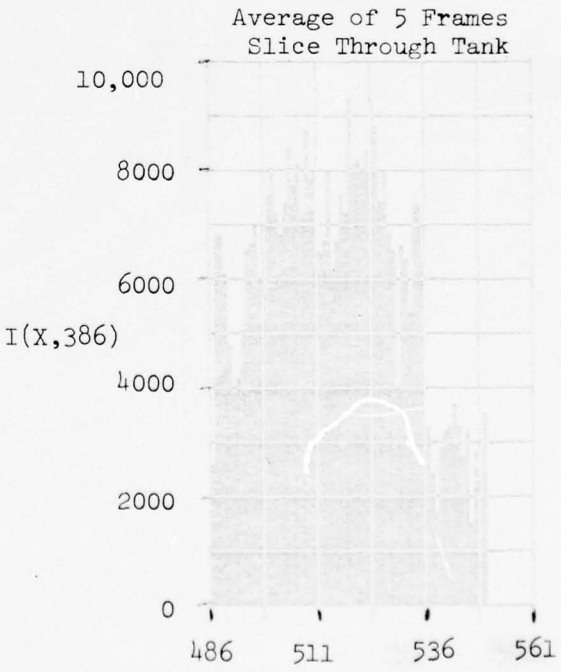
The second set consisted of three nonconsecutive frames. Perspective changes caused objects beyond the vicinity of the matching template to blur badly. Figures 2-32 to 2-37 contain the three frames and enlarged views of the equivalent subelements selected by the cross-correlation routine. Two composite versions are shown in Figures 2-38 and 2-39.

The first composite was obtained from a pointwise average of two image points from the three frames. The two values averaged were those most nearly equal to the mean of all three values. The second composite was obtained by selecting the minimum value of the three image point values from the three frames. Observe that the data marks are almost totally absent in the second composite (Fig. 2-39) because the frame-to-frame offsets were so large and the minimum (i.e., darkest) image value of the three frames was selected for the composite image.

Conclusions:

Interframe averaging improved the FLIR image signal-to-noise ratio (see plots below). However, the available imagery (Figure 2-24) which represented successive TV rames lacked sufficient scene detail to evaluate the perceptual effects. The "busier" scene (Figs. 2-32 to 2-34) imagery did not represent successive TV frames and perspective changes between images badly blurred objects away from the registered subimager.





3.2 Image Enhancement Techniques

3.2.1 Dynamic Contrast Normalization - IMSHADE

A dynamic contrast normalization algorithm (IMSHADE), which has demonstrated its enhancement capability on natural scenes, was the primary enhancement method evaluated with FLIR images. IMSHADE is a compromise between contrast normalization schemes which are based solely on the local or global range of the image intensity function. It generates coarse ceiling and floor arrays from local extrema and then partially "relaxes" these limits toward the scene's global extremes. Consequently, local object-background contrast is enhanced without losing global background variation. Its most impressive results are obtained when global background differences are on the same order as local object-background differences. Enhancement examples of non-FLIR images are found in Figures 2-46 through 2-62.

Many examples of IMSHADE-processed FLIR images are found in Section 2. The poor signal-to-noise ratio and the relatively small global backgound intensity variations in the original FLIR images complicate the evaluation of these results. Nevertheless, the photographs do demonstrate IMSHADES's capacity to increase local contrast without complete loss of global background variations.

IMSHADE Algorithm: (Parameters = N, $100\alpha_C$, $100\alpha_F$, $100\beta_C$, $100\beta_F$, W)

The algorithm's steps are introduced by way of a simplified, one-dimensional example in Figures 3-12 through 3-17.

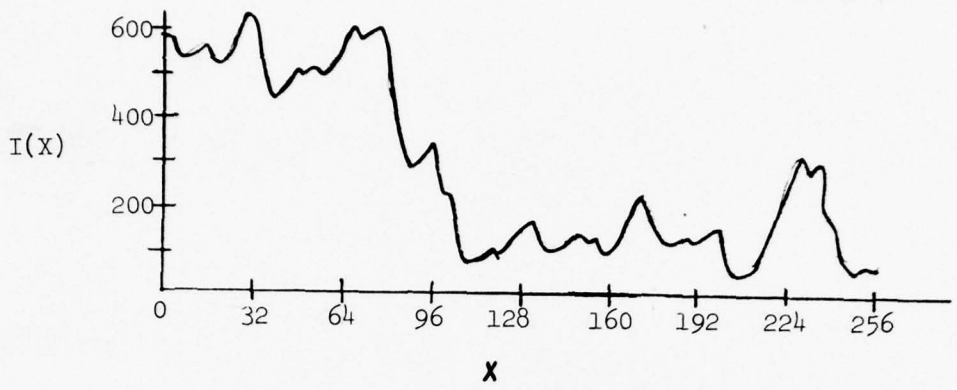


Figure 3-12 - Original Image Function

I(x) is the intensity (or brightness) of a one-dimensional image sampled at 256 discrete points. (The equations to be described are trivially extended to the two-dimensional problem encountered in practice.) If, as is often the case, the size of interesting objects is much less than the full field of view (256 pixels in the example), then what appears to be an edge in Figure 3-12 is probably a change in the background intensity (e.g., the horizon of an IR image). The smaller amplitude peaks and valleys are probably features of greater interest and deserve a greater fraction of the limited display dynamic range than they possess in the raw image. The algorithm steps to be described achieve that goal in a controlled way.

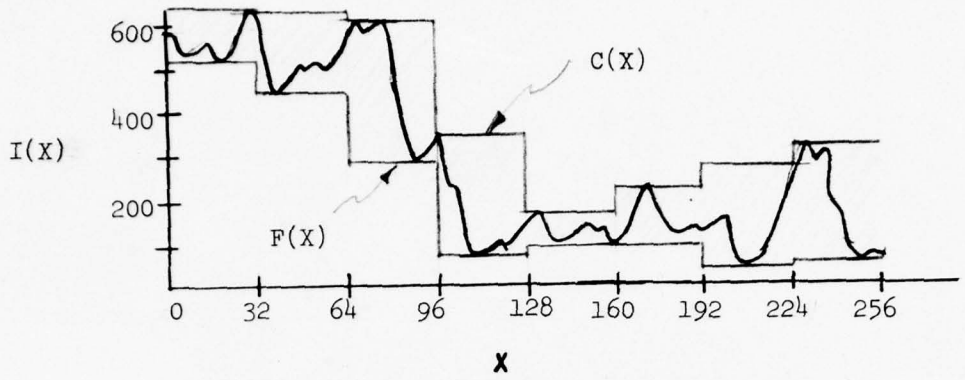


Figure 3-13 - Ceiling and Floor Arrays C(i) and F(i) Defined

Eight "sub-images," each comprised of 32 pixels, are defined. The subimage dimensions are pre-selected to be about the size of interesting features. The intensities of the brightest and least bright pixel from each subimage are extracted and stored in the "ceiling" and "floor" arrays, C(i) and F(i), respectively. The index "i" defines the subimage. Two piecewise constant functions C(x) and F(x) are shown in Figure 3-13 to represent these arrays.

$$C'(i) = \text{MAX}\left\{\frac{\alpha_C \cdot \overline{C}(i)+C(i)}{1+\alpha_C}\ ,\ C(i)\right\}$$

$$\overline{C}(i) = \frac{1}{2}\left[C(i-1) + C(i+1)\right]$$

$$\text{Similarly: } F'(i) = \text{MIN}\left\{\frac{\alpha_F \cdot \overline{F}(i)+F(i)}{1+\alpha_F}\ ,\ F(i)\right\}$$

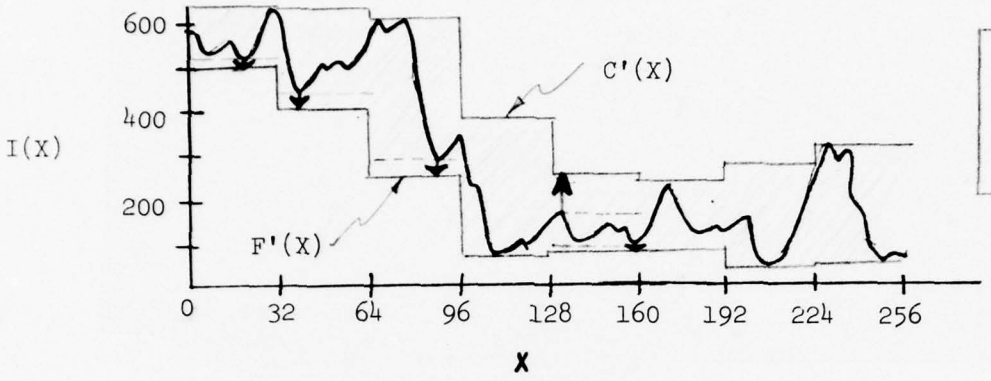
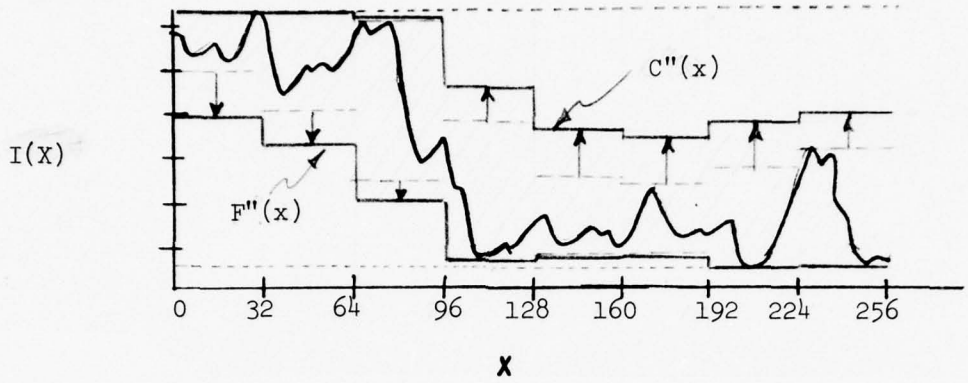


Figure 3-14 - Nonlinear Filtering of C(i) and F(i)

Each extrema value in the ceiling and floor arrays is averaged with the mean value of its nearest neighbors. Coefficients α_C and α_F control the weighting of the two terms in each sum. The nonlinear aspect of the filtering is that an original value is retained if the filtered value does not drive the local extrema towards the global extrema. At the image boundaries, array elements C(0), C(7), F(0) and F(7) are copied into the exterior of the image (C(-1), C(8), F(-1), and F(8)) to perform the average. Finally, the filtering can be performed a variable number of times N, the first argument in the IMSHADE parameter list.

$$C''(i) = \beta_C \cdot C'(i) + (1-\beta_C) \cdot \text{MAX} \{ C'(0), C'(1), \ldots, C'(7) \}$$

$$F''(i) = \beta_F \cdot F'(i) + (1-\beta_F) \cdot \text{MIN} \{ F'(0), F'(1), \ldots, F'(7) \}$$



EXAMPLE:
$\beta_C = .75$
$\beta_F = .75$

Figure 3-15 - Contrast Expansion on C'(i) and F'(i)

The procedure illustrated in Figure 3-15 is the key step in the IMSHADE algorithm. The tight envelope around the image function I(X) formed by the ceiling and floor functions is relaxed. Each value in the ceiling array C(i) is increased a fixed fraction β_C of the difference between its old value and the maximum value in the array. A similar computation is performed on each element of the floor array F(i). This procedure allow the global extrema in I(X) to affect the normalization bounds of every image point. The global extrema are often determined by the range of the variation in the background's intensity, as is the case in our current example, and this step assures that background perceptual effects will be partially retained.

$$C(x) = C(i) + \left[C(i+1)-C(i)\right] \cdot \frac{m}{32} \qquad \text{where } i = MOD(X)$$

$$F(x) = F(i) + \left[F(i+1)-F(i)\right] \cdot \frac{m}{32} \qquad m = R(X)$$

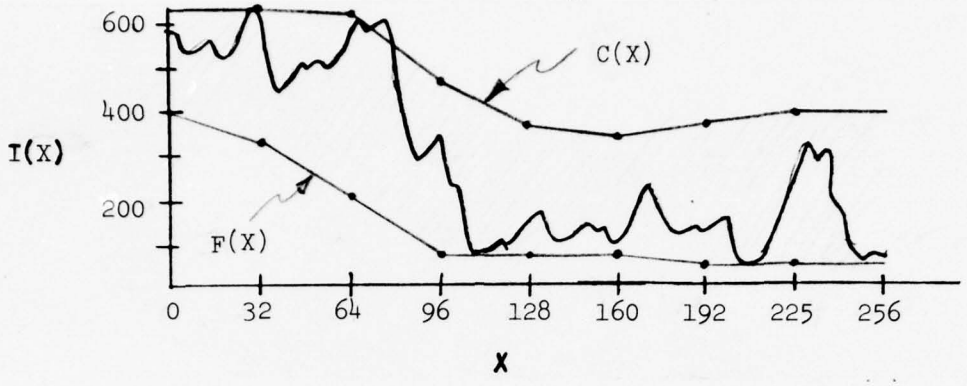
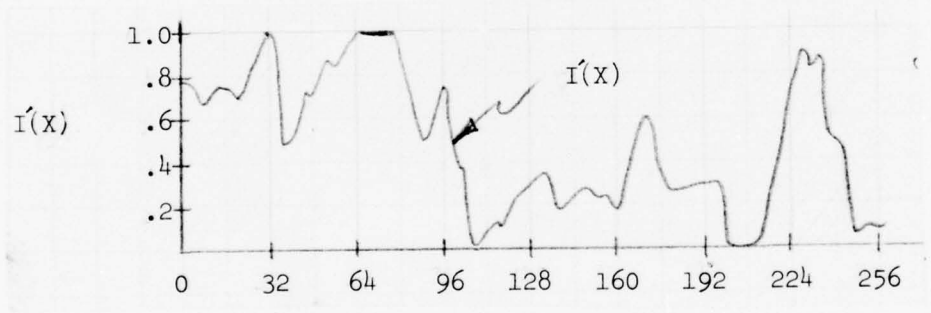


Figure 3-16 - Interpolation of C(i) and F(i)

The final pre-normalization step is to create piecewise continuous ceiling and floor functions. This step prevents the boundaries of the sub-images from showing in the enhanced image.



$$I'(X) = \frac{I(x)-F(x)}{C(x)-F(x)}$$

Figure 3-17 - Pointwise Normalization of I(X)

Each image point value is recomputed in this step. The upper and lower bounds in the normalization equation vary from point to point. This normalized image function, depicted for our example in Figure 3-17, has increased local object-background contrast. The contrast expansion step, however, has assured that the brightest and darkest regions in the original image continue

to be the brightest and darkest regions in the enhanced image. Thus, IMSHADE has achieved a compromise between strictly local and global normalization techniques. The values of β_C, β_F, α_C, and α_F control the tradeoff that must be made in the calculation of normalization bounds.

Extension to Two Dimensional Image Functions

The same steps introduced in the simplified one dimensional example carry over into the two dimensional case. The image is subdivided into 256 subimages. Thus, for a 1024 x 800 image, each subimage is 64 x 50 pixels. Ceiling and floor arrays are once again obtained by extracting the maximum and minimum intensity from each subimage. The two 16 x 16 matrices thereby created form a two dimensional envelope of the image function. Each array is expanded at its boundaries to become an 18 x 18 matrix by copying the original boundary values into the new boundary elements. This step is required for the nonlinear spatial smoothing defined by equations 3.1 to 3.3. The contrast expansion operation which was illustrated in Figure 3-15 is defined in equations 3.4 and 3.5 for the two dimensional problem.

The four point interpolation required to obtain piecewise continuous ceiling and floor functions are defined by equations 3.6 to 3.8 and illustrated by Figure 3-18. The final expression for the normalized image function is given by equation 3.9. The variables W and B select the gain and bias levels appropriate for the display.

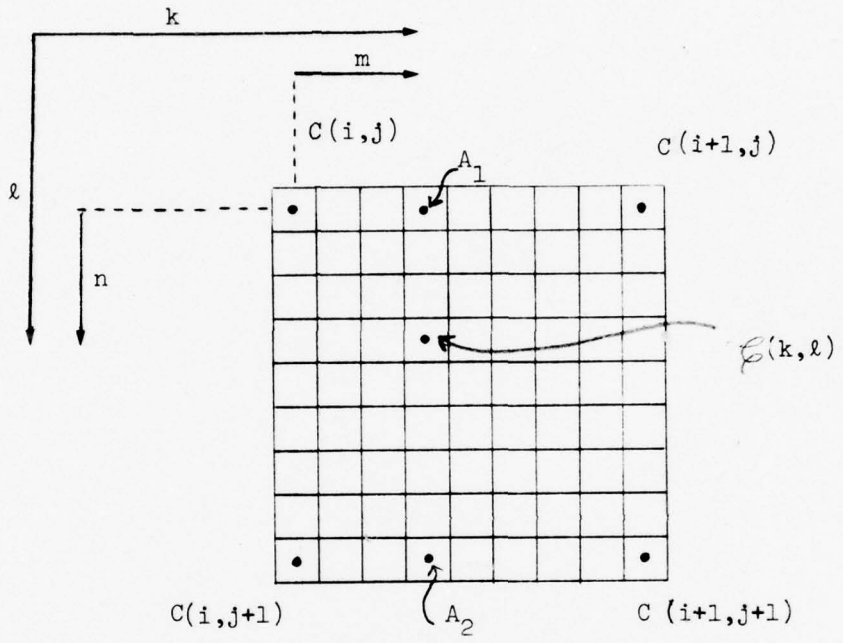


Figure 3-18. Interpolation of Ceiling Array

TWO DIMENSIONAL PROBLEM EQUATIONS

COARSE ARRAYS C(i,j) AND F(i,j) SPATIAL SMOOTHING C'(i,j), F'(i,j)

$$\overline{C}(i,j) = \frac{1}{8}\Big[C(i-1,j-1)+C(i,j-1)+C(i+1,j-1)+C(i-1,j) +C(i+1,j)+C(i+1,J-1)+C(i+1,j)+C(i+1,j+1)\Big] \quad (3.1)$$

$$C'(i,j) = \text{MAX}\left\{\frac{\alpha_C\,\overline{C}(i,j)+C(i,j)}{1+\alpha_C}\ ,\ C(i,j)\right\} \quad (3.2)$$

Similarly:

$$F'(i,j) = \text{MIN}\left\{\frac{\alpha_F\overline{F}(i,j)+F(i,j)}{1+\alpha_F}\ ,\ F(i,j)\right\} \quad (3.3)$$

CONTRAST EXPANSION: C"(i,j), F"(i,j)

$$C"(i,j) = \beta_c C'(i,j) + (1-\beta_c)\cdot \text{MAX}\ (C') \quad (3.4)$$

$$F"(i,j) = \beta_F F'(i,j) + (1-\beta_F)\cdot \text{MAX}\ (F') \quad (3.5)$$

where

MAX(C') = MAX(C'(0,0),C'(1,0),... , C'(15,15))
MAX(F') = MAX(F'(0,0),F'(1,0),... , F'(15,15))

CONTINUOUS CEILING AND FLOOR FUNCTIONS $\mathcal{C}(k,\ell)\,\mathcal{F}(k,\ell)$

i = MOD(k) m = R(k) where MOD ⇒ Modulo 32
j = MOD(ℓ) n = R(ℓ) R ⇒ remainder

$$A_1 = C(i,j) + \left[C(i+1,j) - C(i,j)\right]\cdot \frac{m}{32} \quad (3.6)$$

$$A_2 = C(i,j+1)+\left[C(i+1,j+1) - C(i,j+1)\right]\cdot \frac{m}{32} \quad (3.7)$$

$$\mathcal{C}(k,\ell) = A_1 + (A_2-A_1)\cdot \frac{n}{32} \quad \text{(See Figure 3-18)} \quad (3.8)$$

Similarly, $\mathcal{F}(k,\ell) = A_1 + (A_2-A_1)\cdot \frac{n}{32}$ if F is substituted for C in equations 3.6 and 3.7.

$$I'(k,\ell) = \frac{I(k,\ell)-\mathcal{F}(k,\ell)}{\mathcal{C}(k,\ell)-\mathcal{F}(k,\ell)}\cdot (W-B) + B \quad (3.9)$$

where W and B are the overall display brightness level and intensity bias under control of the operator.

IMSHADE's Variables α and β

The IMSHADE algorithm, as has been explained in the simplified example and detailed equations of the last section, performs filtering on two 16x16 matrices called the ceiling and floor arrays. The parameters α_C and α_F control the degree of spatial smoothing and β_C and β_F control the degree of contrast enhancement. Figures 3-20 through 3-22 illustrate the effects several combinations of settings have on the original cross-sectional plot from Figure 3-19. This example was constructed such that each subimage from the 16x400 full image is 1x25 pixels. This dimension is shown in Figure 3-19.

The most extreme effects achievable with IMSHADE are possible with no spatial filtering ($\alpha_C = \alpha_F = 0$) and full contrast expansion ($\beta_C = \beta_F = 1$). This case is shown for our example in Figure 3-21. Even in the nearly uniformly bright region on the lefthand side of the original image, the full contrast enhancement drives the local minimum down to the global minimum.

This nearly total loss of global background brightness has perceptual disadvantages similar to those of AC coupled FLIR systems.

Most of the examples of IMSHADE found in Section 2 were obtained with no spatial smoothing ($\alpha_C = \alpha_F = 0$) and extensive, but not full, contrast expansion. Figure 3-20 illustrates these conditions for our example. Local contrast is increased, but local minima and maxima do not reach the global extremes reached in Figure 3-21.

Conclusions:

The FLIR imagery available for this study did not have large background intensity changes across the image and therefore did not exhaust the display dynamic range. In fact, increasing local feature-background contrast tended to clutter the FLIR scene away from the target. This effect can be observed from a comparison of the pre- and post-enhancement of a scene in Figures 2-3 and 2-5. It suggests that the setting of the contrast expansion variables β_C and β_F be adaptively determined based upon the dynamic range of the background intensity. Thus, the tradeoff setting between normalization to local or to global extremes performed by the β parameters would be optimized adaptively according to the scene's content. One approach would be based upon the dynamic ranges of the 256 local maxima in the ceiling array and the 256 local minima in the floor array.

LOCAL CONTRAST NORMALIZATION

Slices through a 16x400 image illustrate effect of algorithm's parameter values .

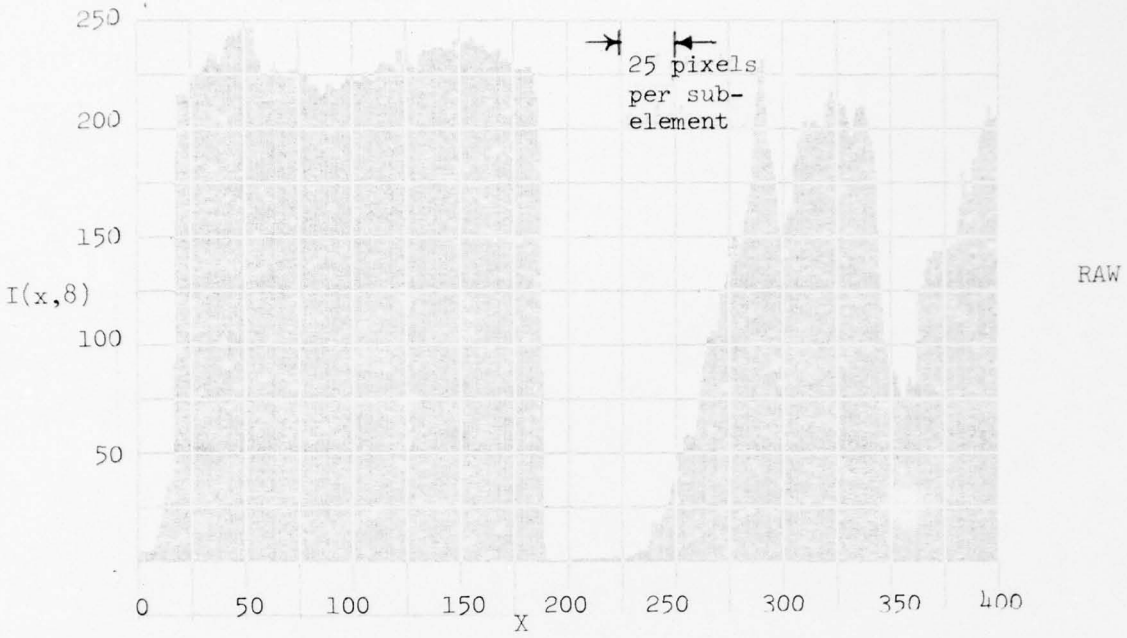


Figure 3-19. Normalized Raw Data (Vertical grid lines divide image into 16 subelements.)

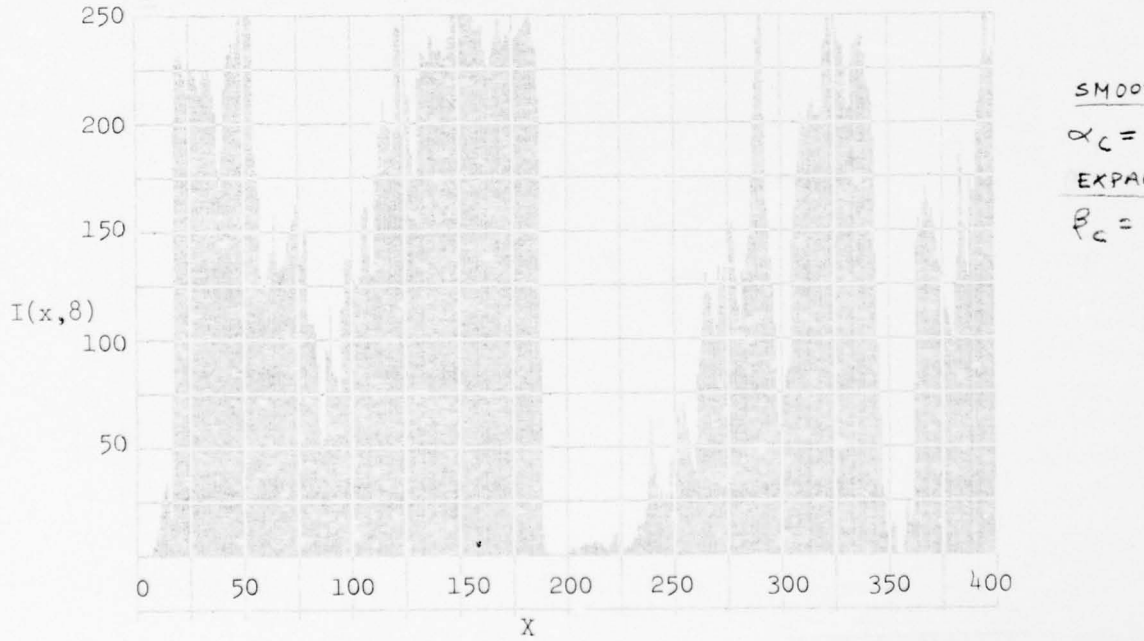
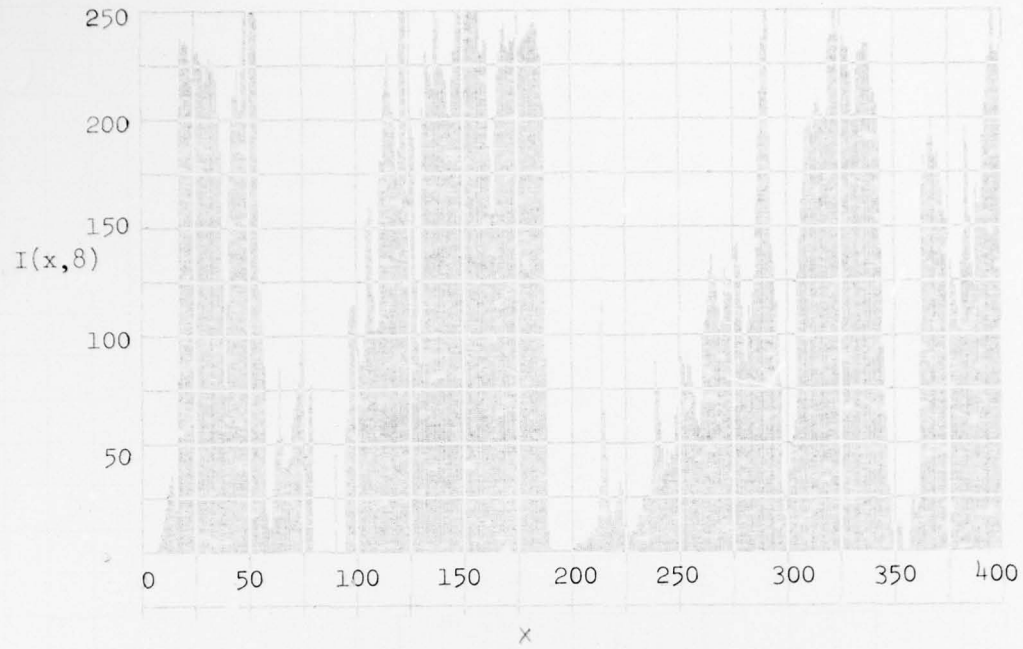
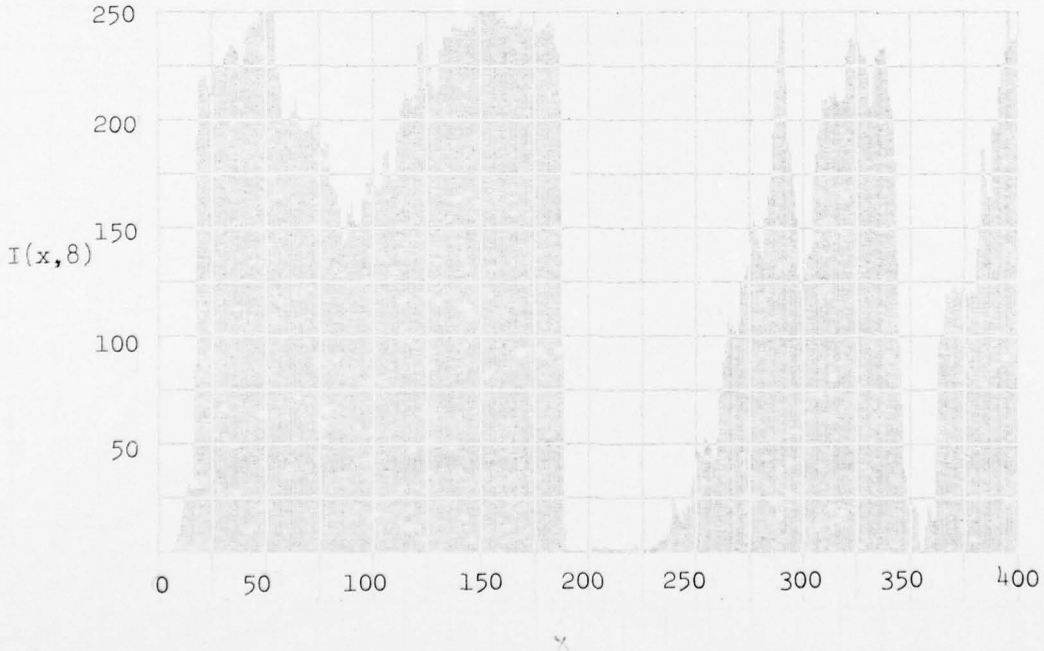


Figure 3-20. Dynamic Contrast Enhancement: $\alpha_C = \alpha_F = 0$ $\beta_C = \beta_F = .75$



SMOOTHING
$\alpha_c = \alpha_F = 0$
EXPANSION
$\beta_c = \beta_F = 1$

FIGURE 3-21. DYNAMIC CONTRAST ENHANCEMENT: $\alpha_c = \alpha_F = 0$, $\beta_c = \beta_F = 1$



SMOOTHING
$\alpha_c = \alpha_F = 1$
(EQUAL WEIGHT TO CENTER PIXEL AND MEAN OF EIGHT NEIGHBORS)
$\beta_c = \beta_F = 1$

FIGURE 3-22. DYNAMIC CONTRAST ENHANCEMENT: $\alpha_c = \alpha_F = 1$ (10 spatial smoothing passes), $\beta_c = \beta_F = 1$

AD NUMBER

PAPER COPY PRICE $

DATE

DDC ACCESSION NOTICE

1. REPORT IDENTIFYING INFORMATION

A. ORIGINATING AGENCY

Rockwell International

B. REPORT TITLE AND/OR NUMBER

Investigation of Image Processing for FLIR

C. MONITOR REPORT NUMBER

J. D Dehne

D. PREPARED UNDER CONTRACT NUMBER(S)

DAAG53-75-C-0245

2. DISTRIBUTION STATEMENT

Distribution Unlimited

INSTRUCTIONS

REQUESTER:

1. *Put your mailing address on reverse of form.*
2. *Complete items 1 and 2.*
3. *Attach form to reports mailed to DDC.*
4. *Use unclassified information only.*

DDC:

1. *Assign AD Number.*
2. *Return to requester.*

DDC FORM FEB 75 50

PREVIOUS EDITION MAY BE USED UNTIL SUPPLY IS EXHAUSTED

3.2.2 Contouring

The FLIR image enhancement potential of contouring can be justified from several considerations. First of all, contouring of hot objects against cold backgrounds has perceptual advantages similar to edge enhancement. Secondly, if hot targets are displayed at near saturation levels, selective contouring of a high temperature band can highlight details that would be imperceptible otherwise.

The image degradation potential of contouring must also be considered however. Targets can easily be hidden among too many contour lines. Low amplitude noise is capable of generating false and partial contour lines, as are many simple contouring algorithms.

Contour Algorithm (Parameters = N, d)

A nearest neighbor algorithm is used to construct contour lines from selected image points. The image points sought are those at the transition of the image intensity function through pre-specified contour levels. The contour levels are uniformly spaced by an input parameter to be referred to as the contour interval (d).

Three steps are required to decide if an image point is selected to be part of a contour line.

Step 1 - Nearest neighbor ranking. The eight nearest neighbors are ranked in order of decreasing intensity. The values of the N^{th} and $(8-N)^{th}$ ranked neighbor are specified as variable "MAX" and "MIN".

Step 2 - Contour level transition determination. The contour levels above which MAX and MIN reside are determined by integer division MAX/d and MIN/d. If these quotients differ, then the image point is a candidate for contour line formation.

Step 3 - Noise filtering. Two additional tests are applied to image points which satisfy Step 2's contour level transition test.

1. If the image point is a minimum or a maximum with respect to MAX and MIN from Step 1, it is rejected as a contour line point.
2. If the difference between the image point and MAX is less than 5% of the contour interval, the image point is not classified as a contour line point.

If a pixel and its neighbors satisfy these tests, then the pixel's image point value is replaced by the largest permissible value.

Application

Attempts to generate contour lines on FLIR images met with very limited success and therefore an evaluation of its enhancement merits is impossible. Figures 2-65 and 2-68 illustrate the results of two experiments. The original FLIR images data underwent local two-dimensional bandpass filtering (Figures 2-64 and 2-67) prior to processing by the contour algorithm. This prefiltering sought to remove noise and to enhance edges. It produced undesirable blotchiness in the background, however, and caused false contour in the contoured representations of the two scenes.

4.0 HARDWARE IMPLEMENTATION

Second generation FLIR systems will perform some signal processing on the focal plane itself. For example, the background suppression function required of all IR imaging systems in order to overcome the enormous D.C. background radiation level will be performed on the focal plane. On focal plane multiplexors will generate a TV compatible video signal. Potential mechanizations of these functions have been developed at Rockwell and reported in a proposal entitled "Second Generation FLIR Image Enhancement," (Report No. T76-257/501).

The need for cryogenic cooling of focal plane electronics, however, restricts the complexity (i.e., power dissipation) of functions integrated with the detector array. Therefore, signal processing functions which accept and generate TV compatible video signals are better suited to off-focal plane mechanizations. The image restoration and enhancement techniques studied under contract fall into this class. Although no detailed analysis of hardware implementations of the nearest neighbor and dynamic contrast normalization algorithms has been done, they possess the attributes desired of inline video processing circuits:

(1) Video input and video output
(2) Real-time performance
(3) Modest storage requirements
(i.e., much less than full frame)

4.1 Nearest Neighbor Implementation

Most of the specific nearest neighbor algorithms evaluated were directed at the particular salt and pepper noise present in currently available

FLIR data, and thus their relevance to next generation FLIR systems is open to question. As a class, however, these methods can perform a wide range of linear and nonlinear filtering operations including gradient based edge enhancement and contouring.

A generic schematic of nearest neighbor algorithm mechanizations is shown in Figure 4-1. Three shift registers, each of which has as many stages as resolution elements in the TV scan line, satisfy the storage requirements. The block entitled "Nearest Neighbor Algorithm Circuit" performsthe specific function desired, such as replacement of cell $I_{k,\ell}$'s value by the average of its eight neighbors if a specific condition is satisfied (e.g., $I_{k,\ell}$ is a local extrema). The time relationship between the incoming and outgoing video is a fixed delay of two scan line periods. Since $I_{k,\ell}$ in Figure 4-1 is simply a filtered version of $I_{k,\ell}$, delayed by two scan periods, it can become the input to another nearest neighbor algorithm circuit or to another video processor such as the dynamic contrast normalization circuit to be discussed next.

4.2 Dynamic Contrast Normalization

The dynamic contrast normalization algorithm is suitable for off focal plane implementations as an in line video processor. Its computation and storage requirements are modest. The generic block diagram for the class of enhancement algorithms to which IMSHADE belongs is found in Figure 4-2. The block entitled "P Parameter Measurement and Storage" monitors the video channel and calculates P regional and global dependent variables on a frame

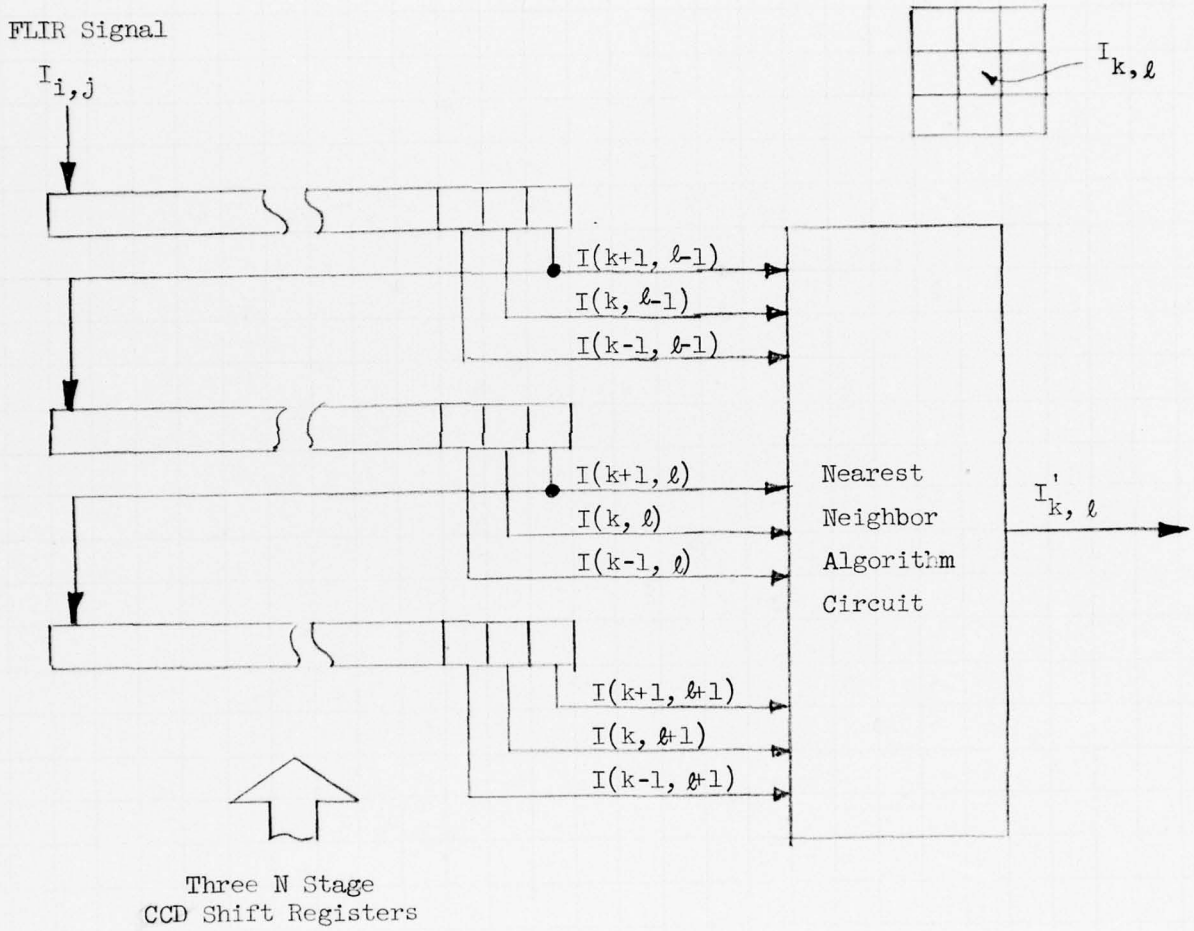


Figure 4-1 Nearest Neighbor Algorithms Implementation

by frame basis. The values computed for the n-1 th frame are used to filter the video in the n^{th} frame. In the figure, k of the P parameters are required by the "Enhancement Algorithm Filter" block to update each pixel value. This block diagram is applicable to any in line video processing technique which can tolerate a one frame delay between measurement of filter variables and their application to the video signal.

A more detailed block diagram of one implementation of the dynamic contrast normalization technique is found in Figure 4-3. The 514 parameters are the global maximum, the global minimum, the 256 subimage maxima, and the 256 subimage minima. The measurement and storage circuitry are shown in block diagram form in Figures 4-4 and 4-5.

Two dimensional interpolation circuitry is required to generate continuous ceiling and floor functions for the final normalization operation. A simple mechanization is described in Appendix A.

4.3 Interframe Averaging Techniques

Inter-frame temporal integration requires that a multiple frame storage register be utilized to store consecutive complete IR frames. In order to realize any MRT improvement at all with this technique requires that more than six IR frames be stored for a 30 fps system (because the eye itself temporally integrates for 0.2 seconds). What are the implications on the required storage capacity? A second generation system will display as many as 10^6 pixels per frame. A factor of 2 increase in MRT at the eye

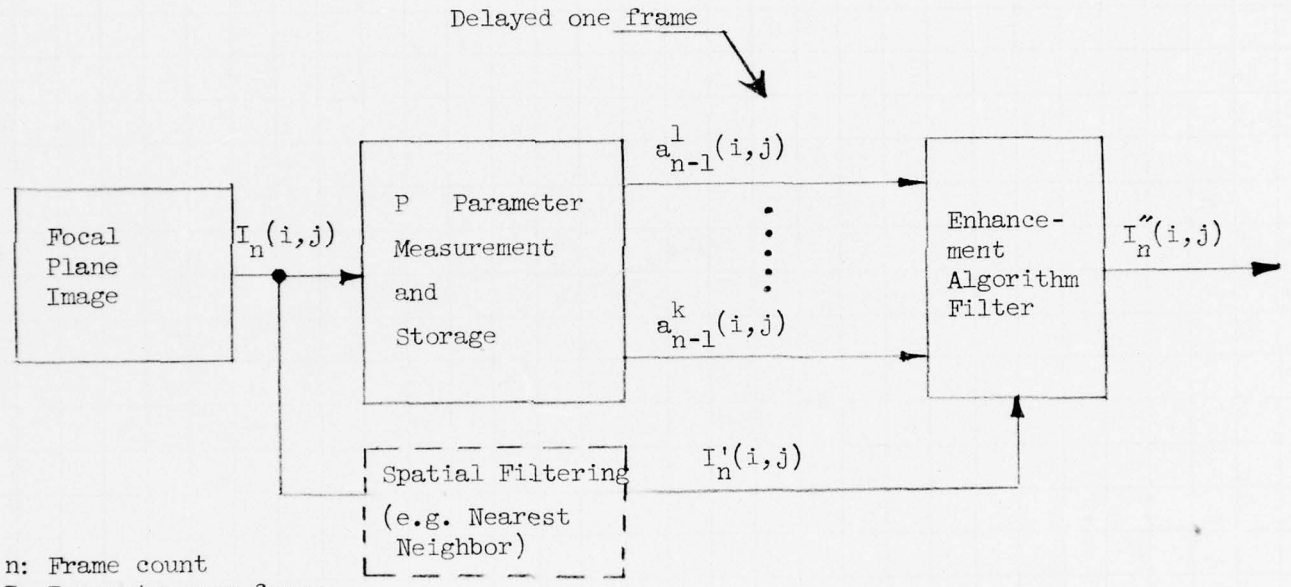


Figure 4-2 Regional/Global Parameter Dependent Signal Processor Block Diagram

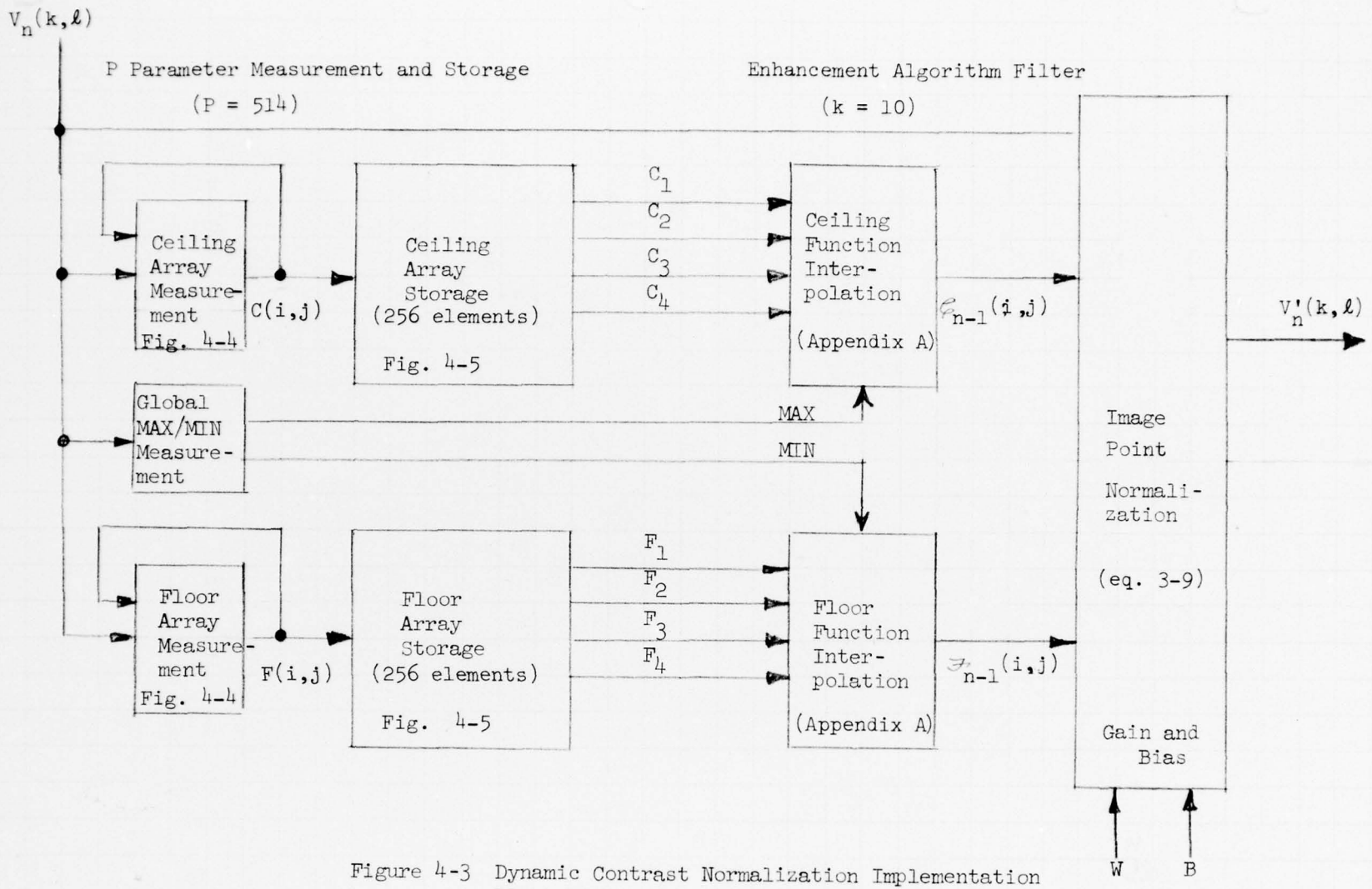


Figure 4-3 Dynamic Contrast Normalization Implementation Block Diagram

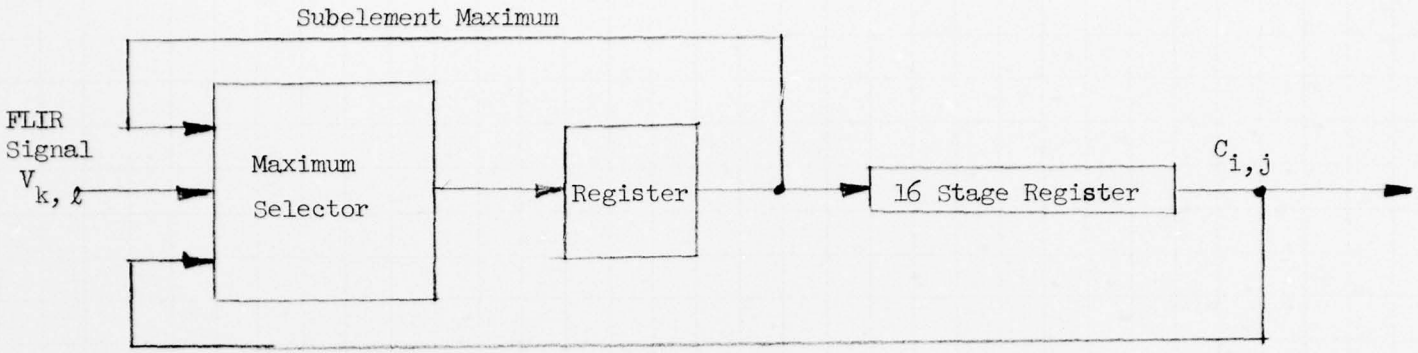


Figure 4-4 Ceiling Computation Circuit (Assumes 512 x 512 Image)
(Floor computation same except minimum selected)

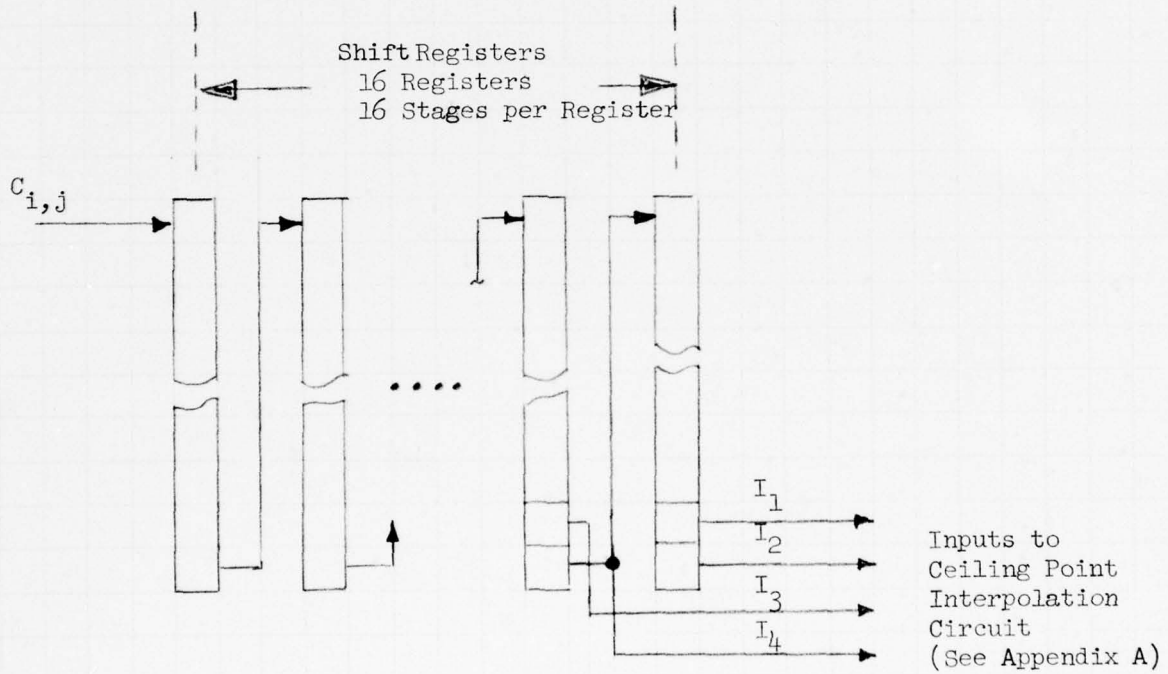


Figure 4-5 Ceiling Array Storage Circuit (floor array circuit is identical)

requires that 24 frames of a 30 fps system be integrated in the storage register. A 100:1 dynamic range would require 7-bits storage capability per pixel. Thus, a digital memory having 1.68×10^8 bit storage capacity would be required for a factor of 2 increase in MRT.

The above discussion assumes a stationary image is being viewed by the FLIR. If 24 frames are being stored and summed before display, then a moving image would cause considerable smear and loss of resolution. This loss of resolution due to image motion would have to be traded off against the increased sensitivity.

5.0 CONCLUSIONS AND RECOMMENDATIONS

The locally dependent contrast normalization technique is an excellent means for freeing the operator from continual in-flight display adjustments. A full evaluation on FLIR imagery was not possible since the available imagery data did not contain sufficient dynamic range (S/N) to exhaust the dynamic range of the display and fully stress the enhancement algorithm.

A significant portion of the test imagery had a target-background to peak-to-peak noise ratio of less than 1. Noise filtering was effectively accomplished by a combination of nonlinear salt and pepper noise removal and frequency domain bandstop filtering. The application of these techniques resulted in S/N ratio improvements of up to 20 db.

The image enhancement and noise filtering techniques investigated are practical and real time implementation.

Significantly improved results in image enhancement, close to those depicted in non-FLIR imagery, Figures 2-48, 2-53 and 2-56, can be expected with advancement in focal plane technology and when signals obtained closer to the focal plane are available.

A three component effort of further image enhancement processing is recommended consisting of: preliminary real-time hardware based enhancement algorithm optimization/evaluation, expanded enhancement (cueing preprocessing) studies by simulation techniques, and systems performance prediction/evaluation study.

The hardware based enhancement algorithm evaluation would serve two functions: first, to provide a firm link between laboratory based image enhancement performance predictions and field performance figures to be used in the full blown performance model, and second, act as a means for collecting data for further image processing research and development efforts.

The expanded image enhancement task should investigate additional processing techniques specifically aimed at FLIR type data as well as applying to, and merging with, the image enhancement and target cueing processing.