



**Calhoun: The NPS Institutional Archive**  
**DSpace Repository**

---

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

---

1979-06

# Implementation of a picture digitizer and display for image processing operations

Costello, Thomas M.; Vogler, Robert C.

Monterey, California. Naval Postgraduate School

---

<https://hdl.handle.net/10945/18753>

---

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

*Downloaded from NPS Archive: Calhoun*



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

**Dudley Knox Library / Naval Postgraduate School**  
**411 Dyer Road / 1 University Circle**  
**Monterey, California USA 93943**

<http://www.nps.edu/library>









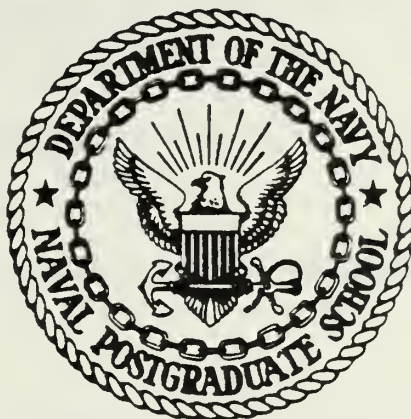






# NAVAL POSTGRADUATE SCHOOL

Monterey, California



## THESIS

IMPLEMENTATION OF A PICTURE DIGITIZER  
AND DISPLAY  
FOR IMAGE PROCESSING OPERATIONS

by

Thomas M. Costello

and

Robert C. Vogler

Thesis Advisor:

June 1979

George A. Rahe

Approved for public release; distribution unlimited.

T189153



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Implementation of a Picture Digitizer and Display for Image Processing Operations		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1979
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Thomas M. Costello Robert C. Vogler		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1979
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		18. SECURITY CLASS. (of this report) Unclassified
		18a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution 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) computer graphics picture digitizer image processing interactive terminal		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The purpose of this study was to implement Spatial Data System's EyeCom Picture Digitizer and Display on Digital Equipment Corporation's PDP-11/50 minicomputer using the real-time operating system RSX-11M. The tasks involved included the following:  - operating the EyeCom on the RT-11 operating system to investigate its image processing capabilities;		



20. (continued)

- implementing of Drexel University's software conversion package for the RSX-11M operating system;
- preparing a EyeCom Operator's Manual for RSX-11M;
- developing a Pseudocolor option using the RAMTEK GX-100A Color Raster Scan Display.

With the completion of these tasks, the EyeCom can now do the following image processing operations: picture digitization, magnification, averaging, gray-scale reduction, contrast enhancement, spatial filtering, shifting, reversal, edge enhancement, picture noise reduction, and histogram distribution. The result of these operations can be displayed on the RAMTEK GX-100A Color Raster Scan Display.





Implementation of a Picture Digitizer and Display  
for Image Processing Operations

by

Thomas M. Costello  
Captain, United States Army  
B.S., United States Military Academy, 1970

and

Robert C. Vogler  
Captain, United States Air Force  
B.S., Butler University, 1973

submitted in partial fulfillment of the  
requirements for the degree of

MASTER SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL  
JUNE 1979



## ABSTRACT

The purpose of this study was to implement Spatial Data System's EyeCom Picture Digitizer and Display on Digital Equipment Corporation's PDP-11/50 minicomputer using the real-time operating system RSX-11M. The tasks involved included the following:

- operating the EyeCom on the RT-11 operating system to investigate its image processing capabilities;
- implementing of Drexel University's software conversion package for the RSX-11M operating system;
- preparing a EyeCom Operator's Manual for RSX-11M;
- developing a Pseudocolor option using the RAMTEK GX-100A Color Raster Scan Display.

With the completion of these tasks, the EyeCom can now do the following image processing operations: picture digitization, magnification, averaging, gray-scale reduction, contrast enhancement, spatial filtering, shifting, reversal, edge enhancement, picture noise reduction, and histogram distribution. The result of these operations can be displayed on the RAMTEK GX-100A Color Raster Scan Display.



## TABLE OF CONTENTS

I.	INTRODUCTION .....	10
II.	THE EYECOM SYSTEM .....	15
	A. VIDICON TELEVISION CAMERA .....	16
	B. DISPLAYS .....	16
	C. DATA INPUTS .....	17
	D. INTERFACE CONTROLLER .....	19
	E. TELEVISION SCANNING OPERATIONS .....	21
	F. PICTURE DIGITIZATION .....	22
III.	IMAGE PROCESSING .....	24
	A. IMAGE PROCESSING FUNCTIONS .....	26
	1. Magnification .....	28
	2. Average Picture .....	33
	3. Reducing Picture Gray Scale .....	39
	4. Histogram and Object Classification .....	42
	5. Contrast Enhancement .....	45



6.	Spatial Filtering .....	49
7.	Reverse and Shift .....	55
8.	Edge Enhancement .....	56
9.	Extracting Vertical and Horizontal Info ..	60
10.	Gradient and Laplacian Method .....	63
11.	Reducing Pictorial Noise .....	68
12.	Reducing Pictorial Data .....	72
B.	PSEUDOCOLOR OPERATIONS .....	74
IV.	CONCLUSIONS AND RECOMMENDATIONS .....	76
A.	CONCLUSIONS .....	76
B.	RECOMMENDATIONS .....	78
V.	APPENDIX A - OPERATOR'S MANUAL .....	81
VI.	APPENDIX B - BASIC OPERATING PRINCIPLES (EyeCom) .	132
VII.	APPENDIX C - RAMTEK PSEUDOCOLOR PROGRAM .....	159
	LIST OF REFERENCES .....	162
	INITIAL DISTRIBUTION LIST .....	163





LIST OF FIGURES

1. THE EYECOM SYSTEM .....	14
2. TELEVISION SCAN PATTERN .....	20
3. ORIGINAL DIGITIZED SCENE .....	27
4. MAGNIFICATION OF TWO .....	29
5. MAGNIFICATION OF TWO .....	30
6. MAGNIFICATION OF 2x4 .....	32
7. AVERAGING 2x2 .....	34
8. 2x2 AVERAGE .....	35
9. 4x4 AVERAGE .....	36
10. 8x8 AVERAGE .....	37
11. 16x16 AVERAGE .....	38
12. 32 GRAY LEVELS .....	40
13. 4 GRAY LEVELS .....	41
14. HISTOGRAM PLOT .....	43
15. FIVE CLASSIFIED LEVELS .....	44



16. CONTRAST ENHANCEMENT FACTOR OF TWO .....	47
17. CONTRAST ENHANCEMENT FACTOR OF EIGHT .....	48
18. SLIDING AVERAGE 3x3 .....	50
19. LOW-PASS FILTER .....	52
20. HIGH-PASS FILTER .....	53
21. ENHANCED HIGH FREQUENCIES .....	54
22. EDGE DATA .....	58
23. ENHANCED EDGES .....	59
24. VERTICAL INFORMATION .....	61
25. HORIZONTAL INFORMATION .....	62
26. EDGE ENHANCEMENT - GRADIENT .....	65
27. EDGE ENHANCEMENT - LAPLACIAN .....	66
28. LAPLACIAN PLUS ORIGINAL .....	68
29. PICTORIAL NOISE .....	70
30. NOISE REDUCTION BY ACCUMULATION .....	71
B1. EYECOM SYSTEM DIAGRAM .....	135
B2. MEMORY PLANE ORGANIZATION .....	146
B3. REGISTER FORMATS .....	150



## ACKNOWLEDGEMENTS

We wish to express our appreciation to Professor Lyle A. Cox for his technical assistance in using the RSX-11M operating system. Additionally, we wish to thank the Computer Laboratory Staff - Bob Limes, Cynitha Irvine, Bill Thomas, and Al Wong for their assistance and perserverance. Also, we appreciated the technical services provided by the Computer Laboratory Technicians - Walt Landaker, Bob McDonnel, and Mike Williams.

Lastly, we wish to thank Professor George A. Rahe, our Thesis Advisor, for giving us the opportunity to use the EyeCom Picture Digitizer and Display and free access to Naval Postgraduate School (NPS) Computer Science Computer Laboratory.



## I. INTRODUCTION

In the past decade with the advent of faster and less expensive computers capable of processing extremely large amounts of data, computer-aided image processing has become an everyday practice. Today, image processing applications are used in medical diagnosis, military target detection, surveillance, and remote sensing of the earth and space. For example, as recently as March 1979, Voyager I's color photographs of Jupiter and its moons demonstrated an instance of what image processing can do with only black and white input photographs.

The purpose of this study was the implementation of Spatial Data System's EyeCom Picture Digitizer and Display with a Digital Equipment Corporation's PDP-11/50 minicomputer using a real-time operating system RSX-11M [5]. Additionally, an interface to the RAMTEK GX-100A Color Raster Scan Display was established to display Pseudocolor digitized pictures transferred from the EyeCom.

The EyeCom Picture Digitizer and Display was added to the PDP-11 system in the NPS Computer Laboratory. This system consisted of two PDP-11/50's and one PDP-11/34 minicomputers, a Macro Arithmetic Processor (MAP-300), and various Graphics equipment, one of which was the RAMTEK Color Raster Scan Display. Two operating systems, UNIX and RSX-11M, are used in the facility, one being assigned permanently to each





of the PDP-11/50's. The RAMTEK Color Raster Scan Display was supported on the UNIX operating system and the EyeCom Picture Digitizer and Display on the RSX-11M operating system. Transfer of data files between the two PDP-11/50's was available.

The tasks involved the following areas: operation of the EyeCom using its component RT-11 operating system on a PDP-11/50 minicomputer to investigate and evaluate the RT-11's integral image processing software programs; testing and implementing a conversion package of the RT-11 image processing software to RSX-11M programs; preparation of a usable EyeCom Operator's Manual for the RSX-11M operating system; and the implementation of Pseudocolor by using the RAMTEK GX-100A Color Raster Scan Display to output EyeCom digitized picture data files.

To complete these tasks, the following areas were considered. The EyeCom had to operate under the available RSX-11M operating system instead of its original RT-11 operating system; consequently, this required implementing the RSX-11M software image processing program package developed by Drexel University Center for Image Processing and Pattern Recognition [6]. A set of general guidelines for utilizing the EyeCom image processing system had to be established in a Operator Manual format similar to the Spatial Data System's Manual on Image Processing Programs for PDP-11 [10]. These guidelines had to be written to instruct the beginning user on how to operate the EyeCom equipment,



how to operate in RSX-11M environment, and how to execute the package image processing programs. To allow for independent user image processing operations, a description of all programs and subroutines used on the EyeCom System was included.

By using the file interface between the RSX-11M operating system and the UNIX operating system, transfer of a RSX-11M picture data file on one PDP-11/50 to a UNIX data file on the other PDP-11/50 was possible. Once the EyeCom picture data file was transferred, a UNIX display program provided the necessary interface to display a Pseudocolor representation of the EyeCom's picture on the RAMTEK Color Raster Scan Display.

In the following chapters, information is provided on the EyeCom's operations, image processing software, Pseudocolor operations, conclusions and recommendations. The Appendices consist of the Operator's Manual, Spatial Data System's hardware description of the EyeCom, and the Pseudocolor Program.

Chapter II covers the EyeCom's major operating principles. An overview of the four functional areas of the EyeCom, the Vidicon television camera; the six displays; the four data inputs; and the Interface Controller is presented.

Chapter III provides a description of the major image processing functions and Pseudocolor operations. For each of the major functions described, actual EyeCom image processing photographs are presented. Also, a description of



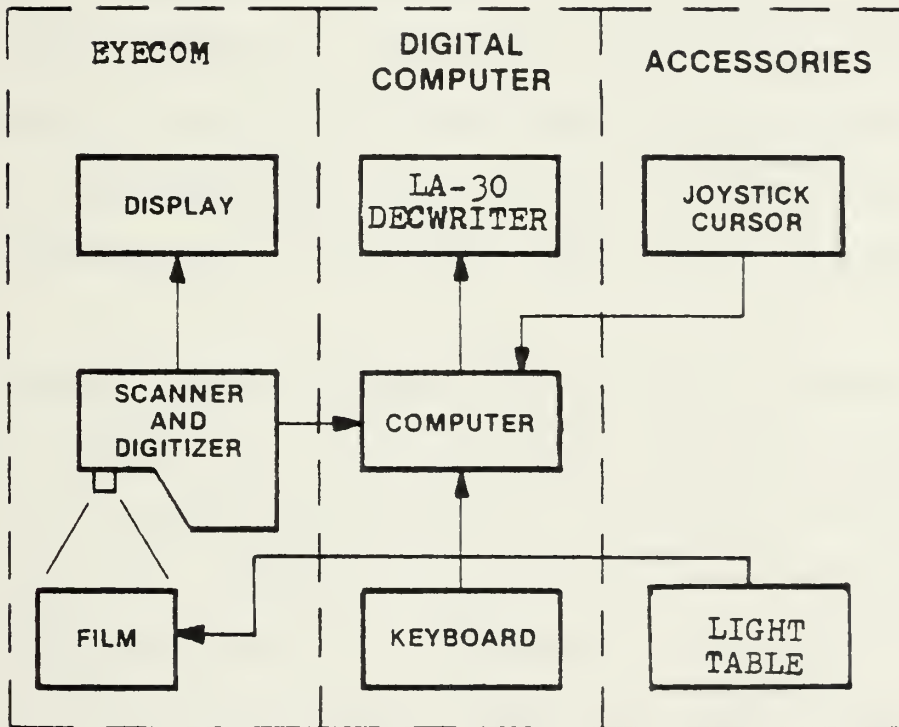
Pseudocolor operations, the computer conversion of a sixteen gray-level picture into color representation is presented.

Chapter IV provides the conclusions and recommendations for future use of the present EyeCom System. Since the purpose of this study was to operate the EyeCom in a real-time environment, all the conclusions and recommendations deal with the results and improvements in that area. The addition of the Macro Arithmetic Processor (MAP-300) will definitely increase the speed of computations in the image processing programs and allow for near real-time operations. By using a separate PDP-11/34, which is available in the Computer Laboratory, the EyeCom would become a separate user system. Finally, to provide a self-contained color image processing capability for the EyeCom, additional hardware attachments were required. These additional attachments are discussed in Chapter III.

Appendix A provides the user with a detailed Operator's Manual that covers how to start operations on the RSX-11M operating system, how to use the image processing programs, how to process a picture, a description of each program and all subroutines used. Appendix B provides Spatial Data System's hardware description of the EyeCom Picture Digitizer and Display. Appendix C is a RAMTEK Program for Pseudocolor operations.



Figure 1. THE EYECOM SYSTEM







## II. THE EYECOM SYSTEM

The EyeCom System is shown in Figure 1 [8]. A minicomputer, a PDP-11/50, receives input data from the scanner (camera) and digitizer. The television camera views the photographs or film transparencies illuminated by the Light Table. The Picture Display allows the user to observe the image being viewed and processed. The Joystick Cursor allows the user to "point" to any pixel on the Picture Display to indicate its position or coordinates to the computer for future use. The Keyboard (LA-30 Decwriter) provides for general communication between the user and the EyeCom System.

The EyeCom Picture Digitizer and Display provides both input and output functions for the processing of images or pictures. The EyeCom consists of four functional areas -- a Vidicon television camera, six displays, four data inputs, and the interface controller. Additionally, the EyeCom combines three types of man-machine computer interaction, i.e., pictorial, graphic, and alphanumeric, into a single unit to do image processing [9].



## A. VIDICON TELEVISION CAMERA

The television camera provides raw picture data to the EyeCom for subsequent digitization and processing. The Vidicon television camera is a black/white camera with a 25-mm f/1.4 lens which has sufficient magnification for most user applications [11].

## B. DISPLAYS

There are six displays used in the EyeCom system. First is the Alphanumeric Display which outputs 24 lines of 80 characters each for operating the image processing programs. Second is the Joystick Cursor Display which may be positioned anywhere on the Picture Display screen using the Joystick control. The coordinates of the Joystick Cursor are available to the computer under program control and used by image processing programs to indicate points on the displayed images. Third is the Graphics Display which is used to draw histogram (graphs) and outline particular areas.

Fourth, the Video Display shows the image being scanned by the television camera. This is the "input image" to the system and is the image that will be subsequently digitized and processed. The picture data (video signal) from the camera is first passed through the analog circuits that can process the picture in real time. The basic analog process-



ing operations are contrast enhancement, gray-scale thresholding, and logarithmic amplification of the gray-scale. The results of the analog processing are immediately visible on the Video Display. Thus, the Video Display shows the image just as it will be digitized.

Fifth, the Picture Display shows images after they are processed by the computer. This display is constantly refreshed from a solid-state random-access memory which can be loaded with digital picture data from the computer. The standard Picture Display presents up to 256 levels of gray in a 640x480 matrix, using a refresh memory contained within the EyeCom display.

The sixth display is the Digitizer Cursor. The Digitizer Cursor, a small white dot, is easily distinguished from the Joystick Cursor, a small white cross. The Digitizer Cursor indicates the point where the digitizer section of the EyeCom is digitizing the picture. Picture digitizing is so rapid that this cursor dot appears as a vertical line as the digitization

### C. DATA INPUTS

The EyeCom provides four methods of inputting data: (1) Keyboard; (2) High-Resolution Digitizer; (3) Real-Time Digitizer; and, (4) Joystick Cursor. Special symbol keys on the EyeCom Keyboard are used in selecting special functions in



image processing programs and for controlling the displays.

The High-Resolution Digitizer provides random access to any of 640x480 points on the Video picture. These digitized points are referred to as picture elements or pixels. The gray-level of each pixel is digitized with a resolution of 8 bits (256 values). The normal sequence starts with the output of an X and a Y address to the EyeCom, followed by a sample command. When the EyeCom has digitized the gray value, an interrupt to the computer is generated and the digitized value is transferred to the computer memory. Usually a large number of pixels are digitized in vertical columns at 15,750 per second. At this speed and using a high performance computer, like the PDP-11/50, a complete picture is digitized in approximately five seconds.

The Real-Time Digitizer provides digitization of the Video picture into 320x240 pixels in 1/60 second. This high data rate, over 12 million samples per second, requires that the digitized pixels be stored in the refresh memory of the Picture Display. Since the refresh memory stores 4 bits for each pixel, the Real-Time Digitizer digitizes each pixel into 4 bits or sixteen gray-levels. The digitizing cycle is initiated by command from the computer and takes place on the next television scan field.

The Real-Time Digitizer is used to provide digitization at high speed and low resolution. The High-Resolution Digitizer provides twice the resolution, four times as many samples, and a much higher gray-scale resolution. Each digi-





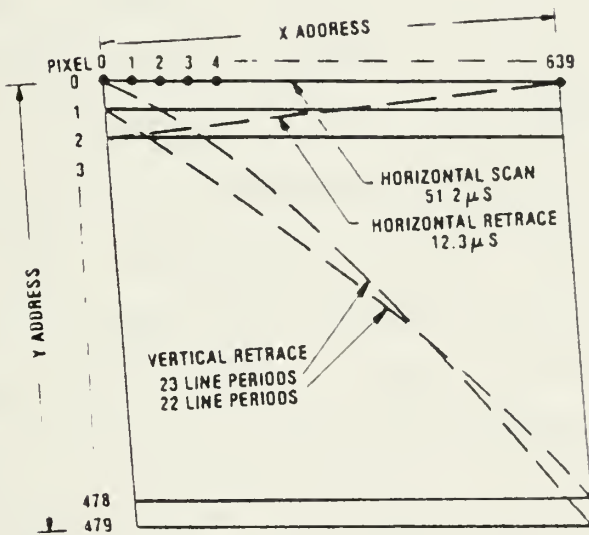
tizer has its own application. The Real-Time Digitizer is used in computer-aided inspection requiring high-speed and rapid image analysis. The High-Resolution Digitizer is used for image enhancements and more detailed measurements where digitizing speed is not critical and high gray-scale resolution is important [9].

#### D. INTERFACE CONTROLLER

The digitization and display of images on the EyeCom generally requires the transfer of relatively large amounts of data at high speed. For example, the High-Resolution Digitizer operates at a nominal data rate of 126,000 baud (bits per seconds). To take advantage of this high speed and to provide the most flexible means of image processing, the EyeCom is operated with a digital computer. A standard controller interfaces the EyeCom to the Digital Equipment Corporation's PDP-11/50 [9].



Figure 2. TELEVISION SCAN PATTERN





## E. TELEVISION SCANNING OPERATIONS

The scanning path covering the image area consists of a series of straight parallel lines (Figure 2) [9]. The spot moves from left to right across each line at a constant speed and then returns to traverse the next line. The spot returns back to the top upon completion of the bottom line. During the horizontal and vertical retrace periods, the exploring and reproducing spots are disabled or blanked so that the image itself is scanned only during the forward, left-to-right motion of the spots.

In order to reduce flicker in the reproduced picture, the scanning lines are interlaced. That is, instead of moving across the horizontal lines in the sequence 0,1,2,3,4,..., the spot first covers the even-numbered lines (0,2,4,...,478) and then the odd-numbered lines (1,3,5,...479). The scanning beam completes two interlaced scanning patterns (fields) to cover the entire image or frame. Each field takes 1/60 second to complete; consequently, the entire image is scanned 30 times a second.

The scanning spot requires 23 horizontal line periods to retrace from the bottom to the top of the picture after the first field and 22 horizontal line periods to retrace the second field. Thus, the scan is repeated after a total of 525 lines (480+23+22) as in standard U.S. broadcast television systems. Since each frame repeats 30 times per second, 525 horizontal lines are sampled 15,750 times per second, or a line every 63.5 microseconds. The spot moves across the



image in 51.2 microseconds and takes 12.3 microseconds to retrace from right to left. The EyeCom samples 640 points across the image. Thus, the finest structure that can be represented by the digital data is 640 alternate bright and dark samples [9].

#### F. PICTURE DIGITIZATION

Pictorial data obtained from the television scanning of a photograph or a negative illuminated by the Light Table, is digitized as follows. The PDP-11/50 minicomputer under the image processing program "MOVE" first loads the X-register and Y-register with the pixel location to be digitized. When the scanning spot of the Vidicon television camera reaches that pixel's X-Y coordinate location on the screen, that pixel is digitized into an 8-bit gray-level value. This intensity or brightness value is then stored in the Z-register where it can be accessed later. Subsequent pixels can be digitized by repeating the above sequence with different X-Y addresses. A detailed hardware explanation of how a picture is digitized can be found in Appendix B.

The digitized EyeCom picture is not a continuous picture like a typical black and white photograph. Instead, it is composed of a number of discrete elements (pixels) each with a unique gray-level (brightness) assigned to it. Because system resolution requires the reproduction of images with a





visual quality that is comparable to the continuous-tone originals, a large number of picture elements (pixels) are needed to give the appearance of spatial continuity, continuous depth, and contrast [8].

The pixel represents the basic element of picture resolution. Each picture is composed of 307,200 unique pixels, divided into an array of 640 columns and 480 rows, stored in 640 memory blocks of 512 bytes each. Each pixel is then assigned a value based on the average level of gray contained in that small portion of the picture it encompasses. The gray scale, or Z-value, of each pixels ranges from 0 to 255, where 0 is black and 255 is white. Thus, all Z-values are available to the computer along with each pixel's X and Y position coordinates [8].



### III. IMAGE PROCESSING

Image processing combines computer applications with modern image scanning techniques to perform various forms of image enhancement, pattern recognition, and object measurements. These operations are then applied to photographic or image format data [8].

Image processing consist of the operations of processing data, manipulation of this data by means of computer application, and producing information out to the user for interpretation. The technology of image processing is based on the fundamental principles which may be similiar to those used by the human being in visual recognition. The human being uses his eyes and brain as the basic tools for visual recognition. The eyes receive the initial data in the form of visual light. Upon receipt of this stimuli, the brain begins its mission of processing the data received by the eyes. The net result is for the brain to interpret the input of the visual light and output an image to the observer. The observer or user can then proceed to act on this image or information perceived.

Image processing has automated this sequence of visual recognition for the user. The system has a television camera as the eyes of the system. The brain is a high speed digital computer. The system then behaves as the human being may in visual recognition. The system can see the image



through the television camera. The programming capabilities of the computer act as the interpreter for the brain. The computer can then manipulate the images received in a number of ways that will lead to the extraction of the desired information.

In general, the two primary end results of the system activities are broken into one of two segments. The first segment consists of an enhanced reconstruction of the original image. The second segment is a numeric or graphic report which relates specific information from the image.

Enhancement involves the sequence of:

IMAGE IN ... PROCESS ... NEW IMAGE OUT

The numeric or graphic report is viewed in a similar manner:

IMAGE IN ... PROCESS ... NUMBERS OR GRAPHS OUT

For either case, it is the extraction of useful information from a raw image data segment in a real-time situation.

The information contained in the Image Processing Functions section were referenced from Spatial Data System's Computer Eye Handbook of Image Processing [8].



## A. IMAGE PROCESSING FUNCTIONS

Any image processing system must be user oriented where the user is not over-burdened by data handling problems and complex mathematical developments. Flexibility of a image processing system is necessary for the user so that it does not detract from the operational goal of picture processing for the user. This flexibility of the digital method in image processing makes a wide variety of linear and non-linear processes possible. The following is a list of possible uses of a system: picture generation, intensity manipulation, geometric manipulation, spatial frequency operation, multipicture analysis, emphasis of details, picture sharpening, image modification, picture interpretation, removal of anomalies, and difference detection between pictures [1].

An example of using some of these functions might involve improving the visibility of features in a photographic image. The process of performing this would begin by reading the image into direct storage from a tape produced by the image scanner. A histogram would show the gray-level distribution. Contrast enhancement would stretch the contrast for the range of gray-levels which contained most of the information. A fourier transform technique could be utilized to generate a digital filter for reducing image blur. Convolution would then carry out the digital filtering. Additional descriptive alphanumeric can be added to the text. Finally, the image can be written back to tape

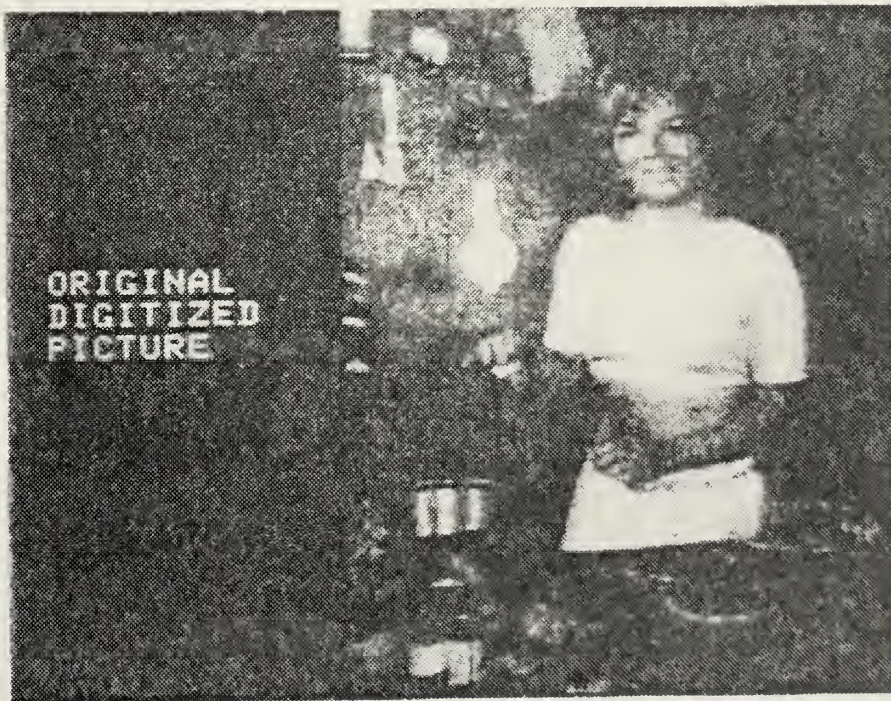




for image displaying.

This is one general application of image processing. Additional applications would include feature recognition by contrast manipulation to accentuate density thresholds, and convolution filtering to detect lines and edges. Figure 3 shows a picture that has been digitized and stored in the computer memory as 307,200 pixels. Each pixel is represented by one of 256 gray-level values.

Figure 3. ORIGINAL DIGITIZED SCENE





## 1. Magnification

The basic principle of magnification is to allow the user to more closely examine a specified portion of a picture. In image processing this function of magnification can be performed by two methods. A zoom lens on the scanner portion of the system can be used to optically magnify the image. The second method of magnification is done by the computer. The software of the system performs the task of magnification. The routine of magnification as done by the computer consists of taking each pixel in the original picture and duplicating this pixel a power of the magnification. If the power of magnification was two, then each pixel would be duplicated four times, a power of two squared ( $2 \times 2$ ). A magnification of four would duplicate a pixel 16 times ( $4 \times 4$ ). Figure 4 is an example of a magnification of two and Figure 5 is a picture of magnification of two.



Figure 4. MAGNIFICATION OF TWO

ORIGINAL PICTURE

0 1 2 4 . . .  
2 3 1 2 . . .  
3 3 1 4 . . .  
4 3 3 1 . . .  
. . . . .

MAGNIFIED PICTURE

0 0 1 1 2 2 4 4 . .  
0 0 1 1 2 2 4 4 . .  
2 2 3 3 1 1 2 2 . .  
2 2 3 3 1 1 2 2 . .  
3 3 3 3 1 1 4 4 . .  
3 3 3 3 1 1 4 4 . .  
4 4 3 3 3 3 1 1 . .  
4 4 3 3 3 3 1 1 . .  
. . . . .





Figure 5. MAGNIFICATION OF TWO



This picture shows a two-time magnification of the original picture.





Since magnification is adding four times the number of pixels to the picture, the size of the new picture as compared to the original picture will only be one-fourth as large. Only one-fourth of the picture will be seen. From an original array of 640x480 pixels, a magnified picture of two will effectively reduce the pixel array by a factor of four to 320x240. With each additional magnification the effective number of pixels will be further reduce. A four-times magnification will have a net effect of reducing the picture to 160x120 pixels, but at four-times the magnification.

Regardless of the number of magnifications performed, the original picture resolution is always maintained. All of the original pictorial data is contained in each magnified version for any image analysis, as corresponding smaller portions of the picture is displayed.

A magnification need not be limited to an even value, i.e., expanding each pixel three times in both the horizontal and vertical directions. The computer can also stretch the picture in the vertical or the horizontal direction. An example of this would be an expansion of a picture by a factor of two in the horizontal direction and a factor of four in the vertical direction or 2x4 expansion.

Figure 6 shows a magnification of two in the horizontal direction and a magnification of four in the vertical direction, or a stretching of the picture.



Figure 6. MAGNIFICATION OF 2X4

ORIGINAL PICTURE

0 1 2 . . . .  
1 3 3 . . . .  
2 2 4 . . . .  
. . . . .

MAGNIFIED PICTURE ( 2 X 4 )

0 0 1 1 2 2 . .  
0 0 1 1 2 2 . .  
0 0 1 1 2 2 . .  
0 0 1 1 2 2 . .  
1 1 3 3 3 3 . .  
1 1 3 3 3 3 . .  
1 1 3 3 3 3 . .  
1 1 3 3 3 3 . .  
2 2 2 2 4 4 . .  
2 2 2 2 4 4 . .  
2 2 2 2 4 4 . .  
2 2 2 2 4 4 . .



## 2. Average Picture

In averaging of a picture, each element is replaced by a mathematical average of an array of pixel values. This averaged value is then inserted back into the array of pixel values. This will give the average value of gray values in a picture. Averaging is used to reduce random variation caused by the grain in the film, the texture of objects, or by scintillations in fluoroscopic images. Figure 7 demonstrates how averaging is calculated for a 2x2 array.

This operation of averaging of pixels can be performed with pixel arrays arranged in a square or rectangular shapes with any number of pixels on a side. In general, the smaller the array of pixels, the lower the amount of loss of picture resolution. In a picture that is averaged with a 2x2 array, the eye can barely distinguish any difference between the original picture and the averaged picture. As the array of pixels increase progressively from 2x2, to 4x4, to 8x8, and to 16x16, the larger picture elements compose the entire image, i.e., a total of only 256 (16x16). The picture will still give the appearance of the original picture if viewed from a distance of approximately 15 feet. At the larger distance, the eye tends to blend the block-like structure together forming a true rendition of the picture. At a normal distance, only this block-like structure is observed. The averaged picture still contains sufficient data for the purposes of extracting image information. Figures 8, 9, 10, and 11 demonstrate this applications.



Figure 7. AVERAGING 2X2

ORIGINAL PICTURE

0 1 3 2 . . . .  
4 7 6 5 . . . .  
7 8 5 2 . . . .  
4 9 8 5 . . . .  
. . . . .

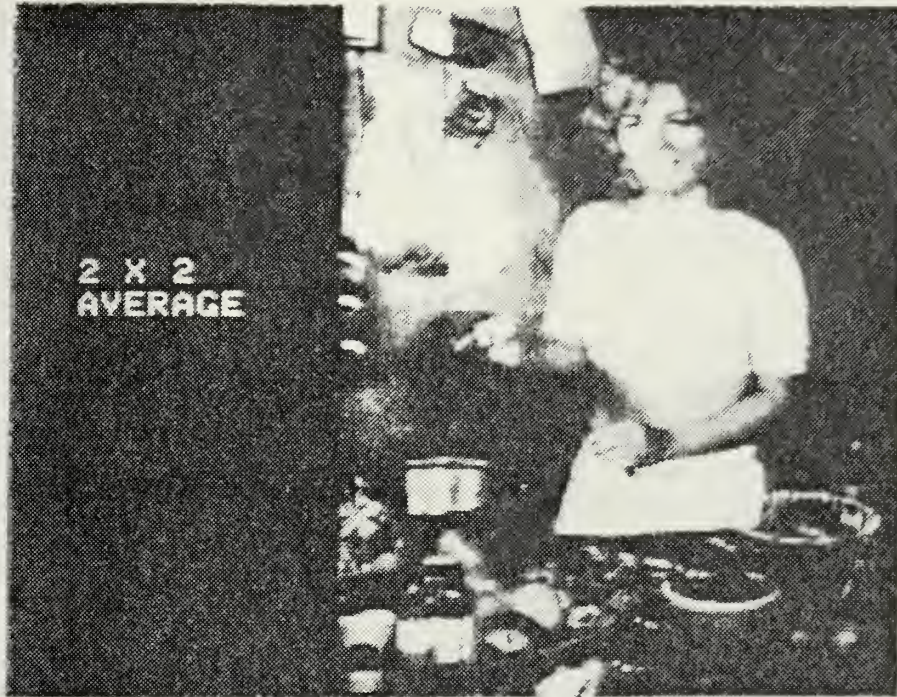
AVERAGED PICTURE

3 3 4 4 . . . .  
3 3 4 4 . . . .  
7 7 5 5 . . . .  
7 7 5 5 . . . .  
. . . . .





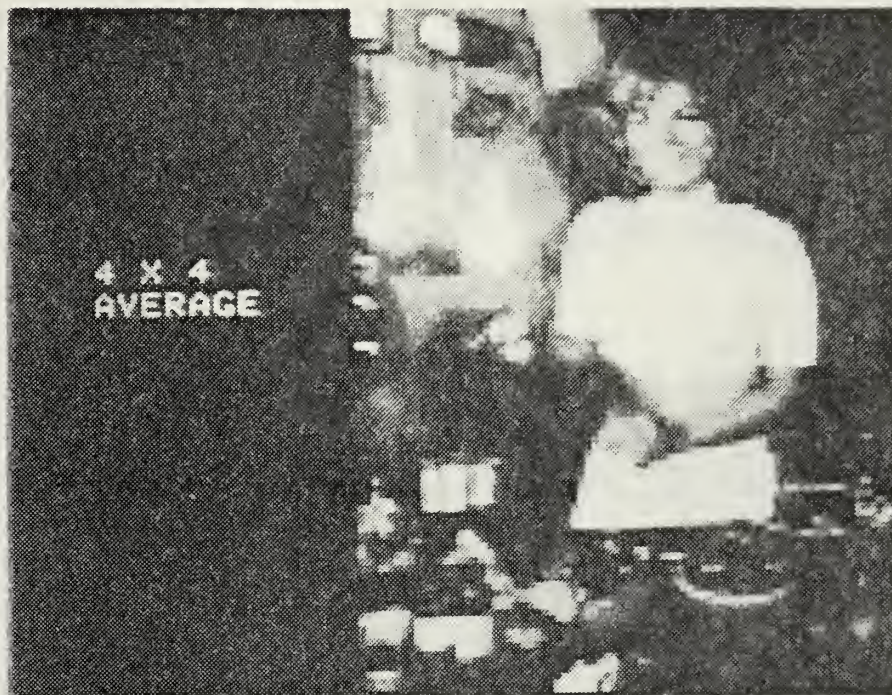
Figure 8. 2 X 2 AVERAGE



From this picture, the eye can barely distinguish any difference between the original picture and this 2 X 2 average picture.



Figure 9. 4 X 4 AVERAGE

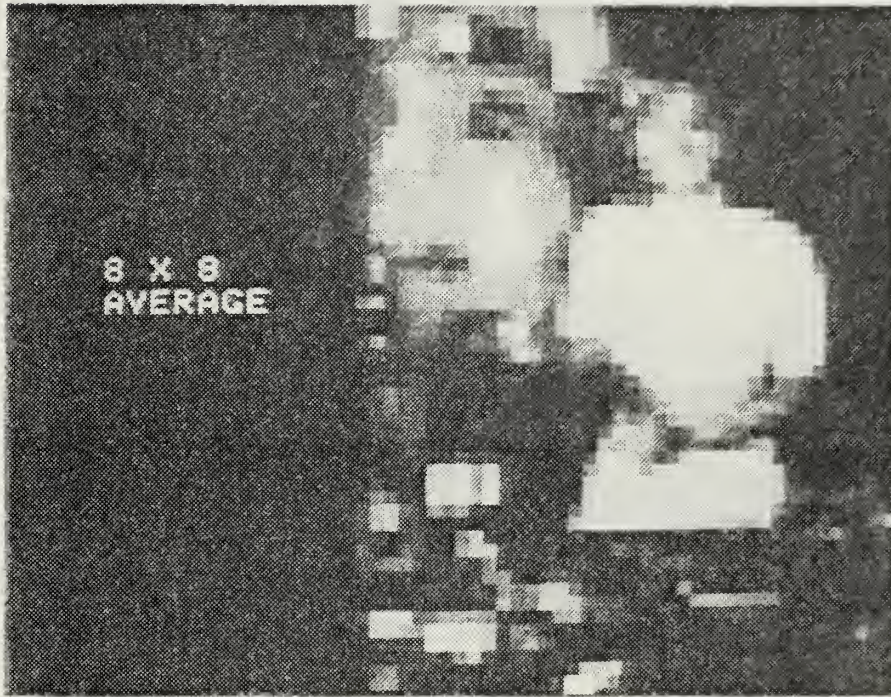


In Figure 9, the picture is beginning to lose some resolution, and approach a block-like structure. The appearance of the scene is still visible.





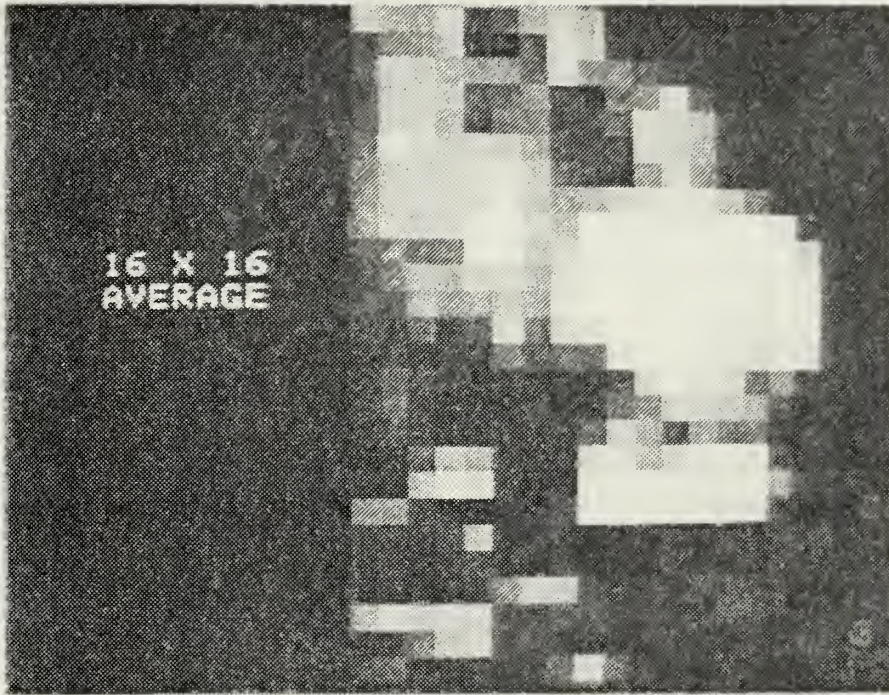
Figure 10. 8 X 8 AVERAGE



In this 8 X 8 averaging, the block-like appearance is readily seen, as well as a distinct loss in picture resolution.



Figure 11. 16 X 16 AVERAGE



A view of this picture at a normal reading distance provides little information. If the picture is viewed at 15 feet, the eye can blend the block-like structure together to give the appearance of the original scene.





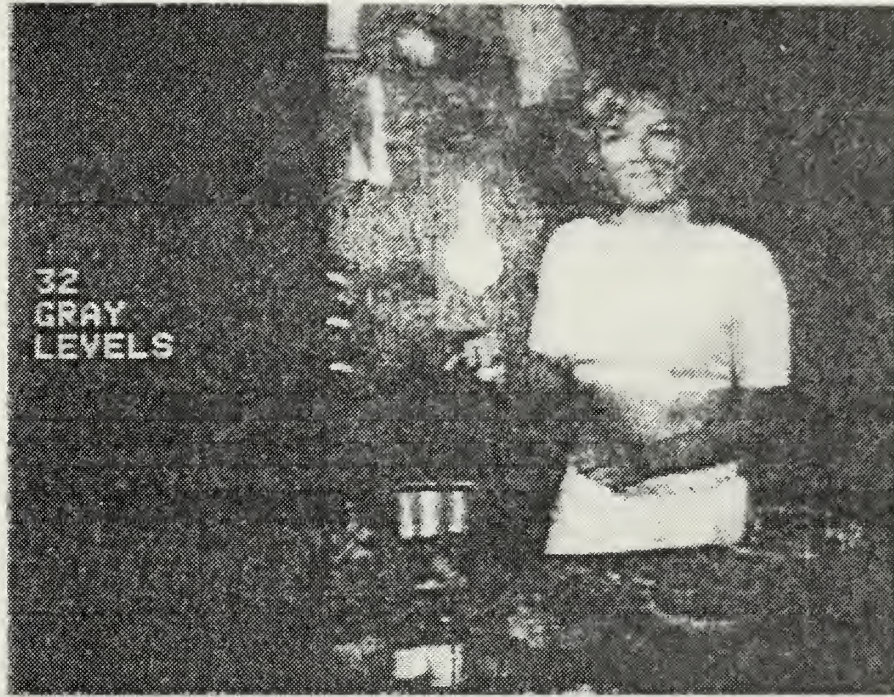
### 3. Reducing Picture Gray Scale

Each pixel is assigned a gray scale value corresponding to a brightness scale. The EyeCom system divides the gray scale into 256 equal parts. With the 256 shades of gray, the reproduced digitized picture appears to the human eye as a continuous picture with gray tones. The reason for this is that the human eye cannot resolve small changes in the gray tone.

Each of the 307,200 pixels has a gray value associated with it. When the number of gray shades are reduced from 256 to 32, the variations of gray from pixel to pixel is much greater than in the original picture. The quality of the picture is barely noticeable. With further reductions of the number of gray-levels, more changes are noticeable. Instead of having continuous shading of gray-levels, specific distinct shades of gray appear that are disjoint. There are noticeable lines that accompany the variations from one gray-level to another. These lines of pixels form lines of equal brightness. The lines of equal brightness enable a user to compare the brightness levels of different parts of a picture. Figures 12 and 13 are examples of this scaling routine, with a gray scale of 32 and a gray scale of 4.



Figure 12. 32 GRAY LEVELS

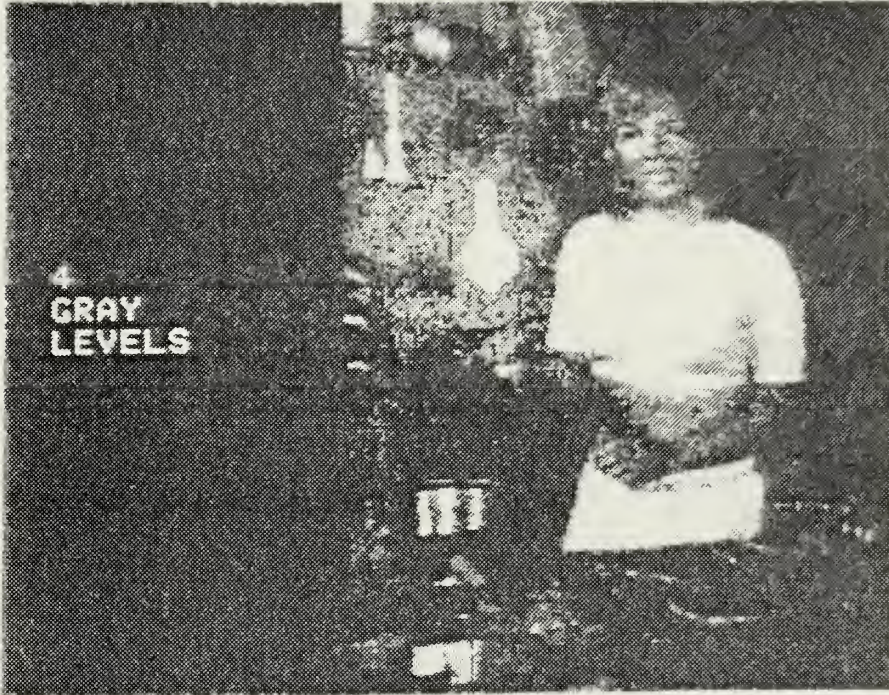


The above figure represents 32 equal parts of a picture on a scale from black to white. The variation is much greater than the original picture but the eye cannot notice the little changes in gray-level change.





Figure 13. 4 GRAY LEVELS



This figure shows the distinct lines of brightness as represented by the four shades of gray.



#### 4. Histogram and Object Classification

A histogram calculates the frequency distribution of gray scale values of a picture. A histogram plot consists of two axes. One axis represents the gray scale, ranging from values of 0 to 255 where the 0 value stands for black and the value 255 is for white. The second axis represents the total number of pixels in the picture. The curve or plot then shows the total distribution of pixels for each shade of gray.

Using the histogram plot, the computer can reconstruct the picture by using the prominent features from this plot. The plot can be broken into any number of unequal shades of gray using the gray scale from the plot. The resulting picture from these classification of gray-level values clearly delineates the features presented. Further functions can show the percentage of area, measure average brightness, or enhance details.

Figure 14 shows the histogram plot of gray scale along the horizontal axis, and the number of pixels for each gray scale along the vertical axis. By using the joystick cursor, individual gray level values can be sampled. Figure 15 is the result of dividing the gray scale into five unequal parts.





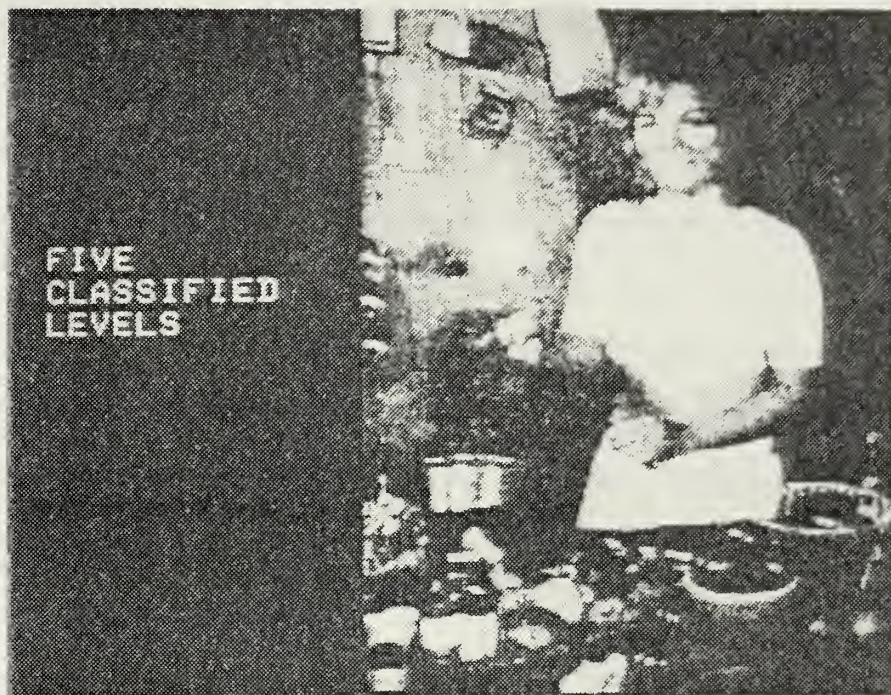
Figure 14. HISTOGRAM PLOT



The plot above shows the distribution of pixels ranging from black to white (left to right). Additional functions of the histogram program can give specific values of this plot and sample any individual pixel on the picture for specific gray-level values.



Figure 15. FIVE CLASSIFIED LEVELS



This picture is a result of building a partition table from selecting specific areas of the histogram plot and dividing the picture into five unequal parts to produce a new picture.





## 5. Contrast Enhancement

The operation of contrast enhancement enhances the differences in the gray tones of a photograph by determining the difference of brightness between adjacent parts. In a high contrast picture, the human eye is sensitive enough to be able to see the picture more readily. The concept of contrast, where the picture is determined by the magnitude of brightness difference between adjacent parts, can be used to render low contrast images more readable and to bring out details that appear washed out in ordinary pictures.

The operation of contrast enhancement increases the contrast of a picture by expanding the gray scale. An example of a low contrast picture with gray values in a mid-range between 50 to 200 is a picture devoid of bright and dark highlights. This picture would give a washed out appearance to the observer. The enhancement algorithm expands the contrast range from the original span of gray value to a new span of gray scale values using a mathematical relationship. The algorithm consists of subtracting a constant value from each pixel, in this example 50. The algorithm then multiplies each pixel by a factor determined by parameters, normally a ratio of new gray values to old gray values. A constant is then added which increases the brightest pixel and the white zone to 255. All pixels in this high zone are now displayed as white in the enhanced image. This gives the effect of stretching the gray scale from a relatively narrow range to a full contrast image with



the full range of gray values.

A typical contrast enhanced image will provide for a blacker dark area and a whiter light area than in the original picture. This would give a high contrast appearance.

The multiplier, the ratio of new gray value range to old gray value range, can be increased such that the contrast is also increased. This is accomplished by forcing more pixels into the white zone. The higher the multiplier the more of a completely black and white image will result with only a few high contrast gray areas. Figures 16 and 17 represent a contrast enhancement with a factor of two and eight respectively.

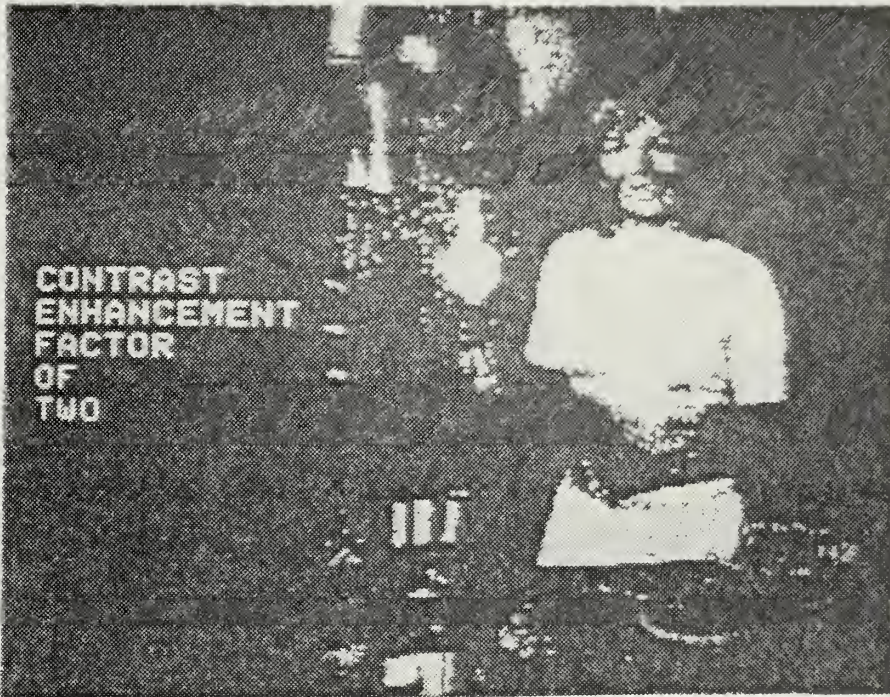
Further contrast enhancement of a picture can result in a picture being devoid of gray areas. By displaying all pixels above a middle value as white, and all pixels below this middle value as black, the resulting picture will consist of black and white images and no middle gray values.

In addition to these functions, contrast enhancement may be used to enhance the darker areas of a picture and, at the same time, change the dark areas to white to show the details of the original dark areas. This is an example of reverse contrast enhancement. In each of these operations, different details of a picture can be focused on to extract specific information for future use.





Figure 16. CONTRAST ENHANCEMENT FACTOR OF TWO



A contrast enhancement has been applied to the original picture using a factor of two. The black areas have become darker and the white areas have become lighter.



Figure 17. CONTRAST ENHANCEMENT FACTOR OF EIGHT



Using a factor of eight for this picture, more pixels are forced into the white zone giving a more black and white appearance than in Figure 16.





## 6. Spatial Filtering

Spatial filtering is defined as the number of changes per inch in gray scale across the picture. Spatial filtering is a computer operation which allows a picture to be separated into high and low spatial frequency components. A low spatial frequency picture or area of a picture would consist of an area whose brightness is virtually the same level throughout this area. A high spatial frequency area would exhibit a varying brightness levels.

In spatial filtering, the computer operation allows a picture to be separated into high and low spatial frequency components. In a low pass filtering operation, the high spatial frequency interfering lines in a picture are eliminated. High pass filtering can be used to enhance picture details by removing the low spatial frequency.

A sliding average value is calculated for every pixel in the picture and then substituted for the original value to obtain a low pass filtered picture. This average can be calculated using a rectangular or square array of any number of pixels surrounding the pixel in operation. Figure 18 demonstrates a 3x3 sliding average to produce a filtered picture.



Figure 18. SLIDING AVERAGE 3X3

ORIGINAL PICTURE

0 1 3 4 7

2 5 4 9 7

8 3 1 6 4



FILTERED

PICTURE

3 4 5





The average value of the 9 pixels in the array replaces that pixel with the new value calculated. The operation then performs the same operation for the next 9 pixels to get another new pixel value. This is done for each of the 307,200 pixels.

This operation will eliminate all high spatial frequency details, leaving only low spatial frequency areas. The high spatial frequency details are those details which alternate between light and dark areas, whereas, in the low spatial frequency details the brightness areas are virtually the same. A high pass filtered image can be obtained by subtracting the low spatial frequency picture from the original picture. By doing a contrast enhancement operation on the high spatial frequency picture, the contrast of the picture can be expanded. Details of this new picture are now enhanced. The combination of spatial filtering and contrast enhancement is an example of combining two operations to produce an effective picture. Figures 19, 20, and 21 highlight the use of spatial filtering functions.



Figure 19. LOW-PASS FILTER

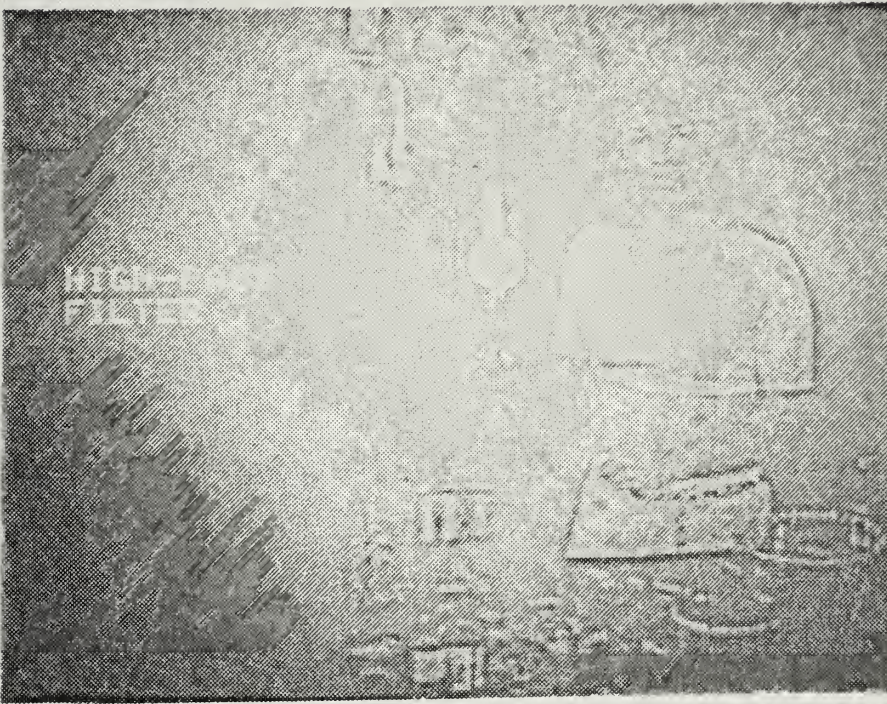


This figure is the result of removing the high spatial frequency details from the picture which were obliterated by the filtering. The low spatial frequency areas are retained.





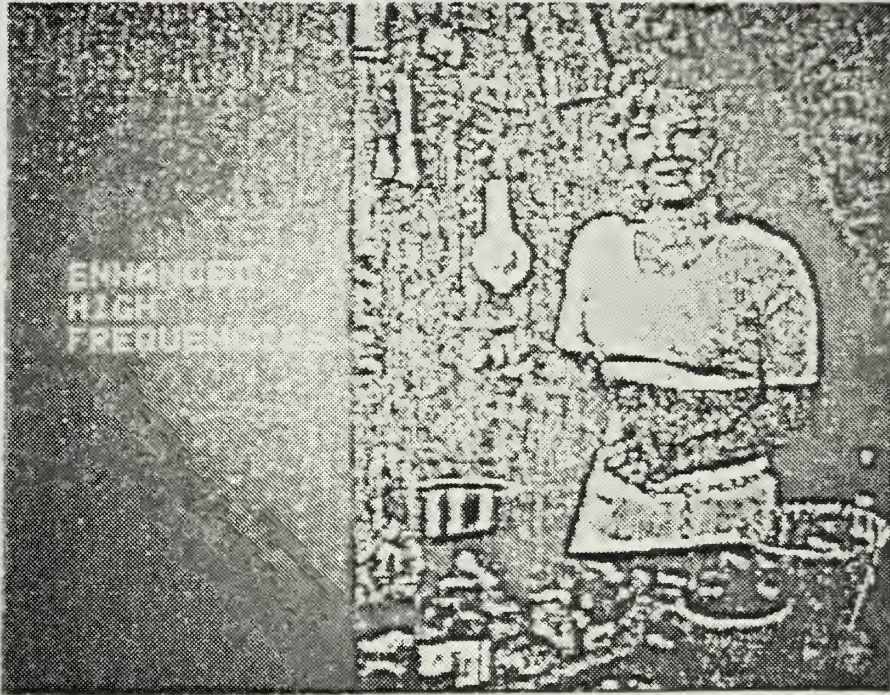
Figure 20. HIGH-PASS FILTER



By subtracting the low pass filtered picture from the original picture, the resultant image shows the high spatial frequency details that were removed in the low pass filtering.



Figure 21. ENHANCED HIGH FREQUENCIES



A contrast enhancement has been performed on the high pass filtered picture (Figure 20) to enhanced the details.





## 7. Reverse and Shift

Operations as reverse and shift serve as functions that are used principally as instruments for subsequent, more utilitarian operations. These routines allow further manipulation of the picture by additional computer operations to produce a more visual picture to the user.

Reversing the gray scale produces a negative image whose visual effect places a different emphasis on the picture. Image reversal is a contributory function to the subtraction of one picture from another. This operation can be applied to additional image processing applications, such as high pass filtering and reducing picture noise.

Shifting a picture provides additional image processing functions that can enhance an image. The shifting of pixels up, down, left, or right can provide functional tools for analyzing an image, such as in edge enhancement.



## 8. Edge Enhancement

Edge enhancement is the process of obtaining information from a picture by looking at the edges surrounding an object in a picture. With edge enhancement operation, the shapes and details of an object or picture become more visually conspicuous and easier to analyze.

A pictorial edge is simply a sharp change in the gray-levels between two or more adjacent pixels, i.e., light to dark or dark to light. This is what the eye can easily detect in an edge enhancement function. The sharp difference in adjacent pixels can be in a horizontal or vertical direction as well as in both directions. The horizontal edges are formed by a string of pixels in a horizontal direction having different values than those pixels either above or below them. The same is true for a vertical string of pixels when compared with pixels to either the left or right. The difference between the two results in the edge enhancement.

The processing of detecting this edge difference consists of shifting the original picture one pixel element in one direction. The next step is to take the difference between pixel values of the original picture and the shifted picture. The result of this difference can be one of three. A middle gray value is substituted for the pixel if there is no difference between the two pixel values. A darker value is substituted if there exists a negative difference, and a lighter value is obtained if a positive difference results.



The total result produced is a white line for a transition of light to dark, a black line for a change of dark to light, and an overall image consisting of a gray background. The black and white lines will be surrounding the edges of all the objects.

To get an enhancement picture in both horizontal and vertical direction at the same time, it is necessary to shift pixels in both directions by one pixel, i.e., up and to one side or down and to one side. The difference values are then generated along both axes. This gives edges for the picture in both directions. Figure 22 is an example of using the shift routine to obtain an enhanced picture along both the horizontal and vertical edges. The picture was shifted in both directions by one pixel.

Additional operations can be performed on this enhanced image. Normally, a contrast enhancement will be required to draw out the contrast range. This will result in having all the edges and details show more prominently as black and white lines against a monotone gray background. Figure 23 is the result of performing a contrast enhancement on Figure 22.





Figure 22. EDGE DATA



This picture is the result of shifting pixels up and to the left, and then subtracting this result from the original picture. The outline of the edges of the picture can be seen here.





Figure 23. ENHANCED EDGES



To get a better visual interpretation of edge data, a contrast enhancement has been utilized on Figure 22 to produce this picture. The edges of the picture can be seen more clearly now as compared to Figure 22.



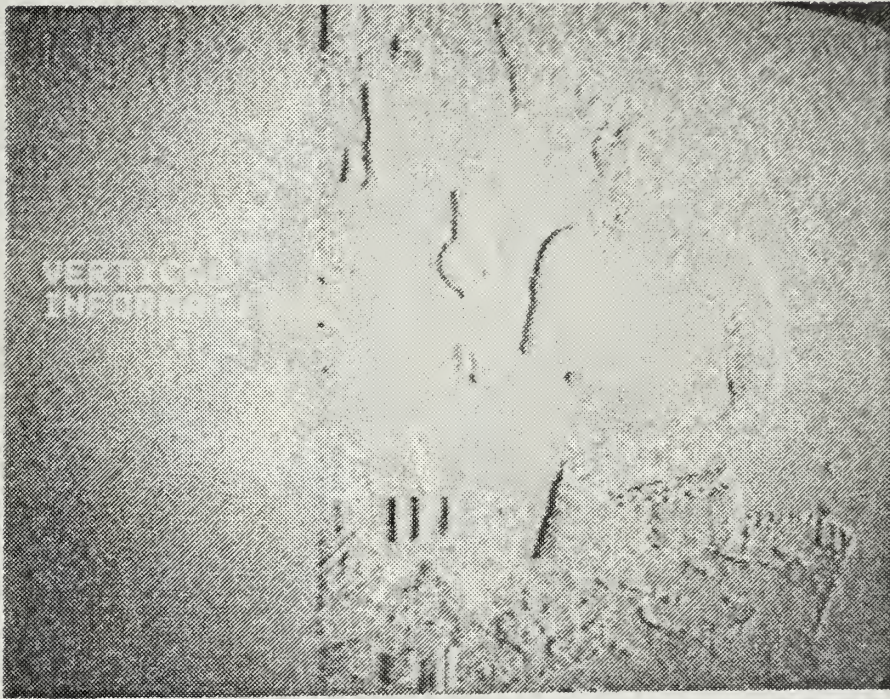
## 9. Extracting Vertical or Horizontal Information

The extracting of just vertical or horizontal information is a spin-off from edge enhancement function. For this case only one axis is necessary to work on an image for either vertical or horizontal information. In edge enhancement, it was necessary to shift a picture in two directions to enhance both horizontal and vertical edges and perform the difference method. When the picture is shifted in only one direction and a difference is performed on the image, only one axis appears in the enhanced picture. The edge line along the axis that was shifted will appear and the opposite axis will have that edge line eliminated from the image. This operation can be very useful when interfering lines appear vertically or horizontally on a picture. Figure 24 is an example of extracting information along the vertical direction and Figure 25 is a picture with horizontal information.





Figure 24. VERTICAL INFORMATION

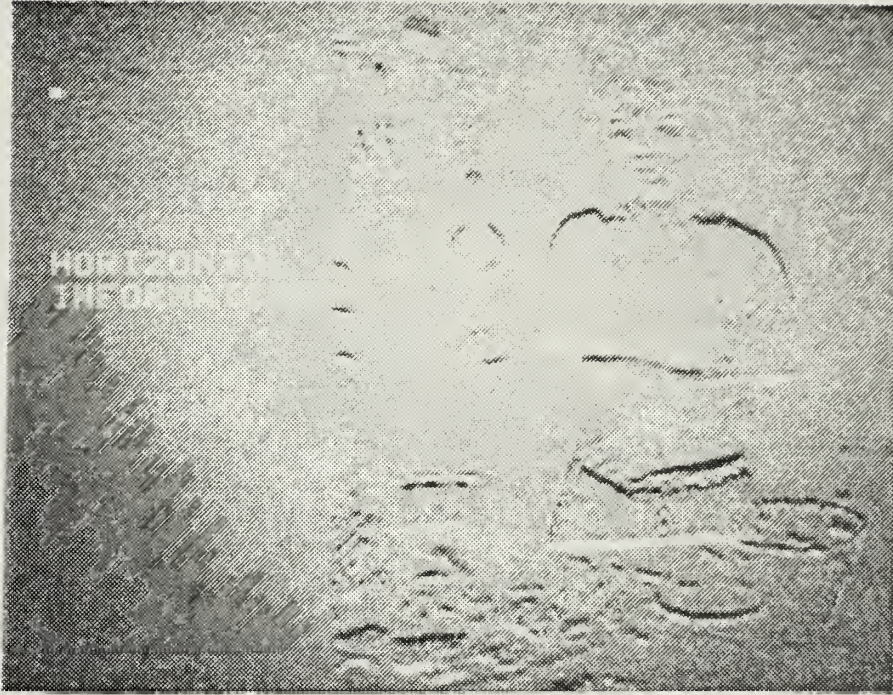


This figure was generated with a shifting of pixels to the right, and then subtracting this result from the original picture. All the horizontal information has been eliminated from this picture.





Figure 25. HORIZONTAL INFORMATION



For this picture, the pixels were shifted up, with the result being subtracted from the original picture. The vertical edges have been eliminated in this picture.



## 10. Gradient and Laplacian Method of Edge Enhancement

A gradient method of edge enhancement image is a further refinement of the edge enhancement function where only the difference of pixel values was taken into account. The gradient method calculates the difference between adjacent horizontal and vertical pixels as was done in the edge enhancement function. These difference pixel values represent the rate of change, or the first derivative of gray-levels across the picture. The gradient method is then calculated as the square root of the sum of the squares of vertical and horizontal derivatives. The edges obtained for this operation are calculated without regard to direction on the image as done in the edge enhancement function. The resulting image portrays all details as white lines on a dark background.

The edge enhancement by the Laplacian method is accomplished in a similar manner. The Laplacian method is based on the second order derivative. This is the sum of the second partial derivative taken in the vertical and horizontal direction. The main difference between the gradient and Laplacian method is that the Laplacian method emphasizes maximum values or peaks within the image. This representation is a more natural way of viewing the image with the eye, with emphasis on the peak values.

The Laplacian method can be carried further with additional computer operations. If a contrast enhancement is performed on the Laplacian image, a black and white binary



picture is produced. This resulting image shows all edges as intensified white lines against a black background. Also, a higher degree of enhancement can still bring out additional details in the image that would not ordinarily be seen with just the Laplacian image.

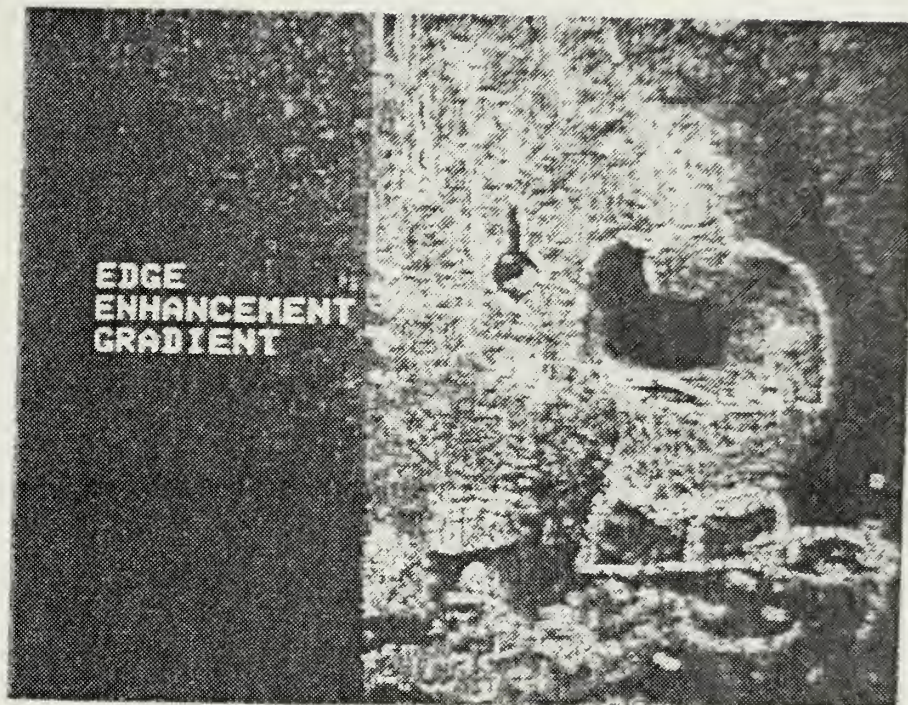
A second way of producing further details in the image is with the original picture itself. When the Laplacian image is added to the original picture, the resultant image has an apparent increase in resolution. The actual picture resolution is still 640x480, but the amplified edges from the Laplacian method accentuate all objects in such a way as to cause the eye to see an increase of the picture. This increase of the picture is due to the attractiveness of the image which the eye sees which would ordinarily be attributed to a greater picture resolution. This example demonstrates the correspondence between the physiological characteristics of the human eye and the computer operation performed on the image.

Figure 26 shows the result of using the gradient method to enhance a picture, and Figure 27 uses the Laplacian method. In Figure 28, the original picture has been added to the Laplacian picture.





Figure 26. EDGE ENHANCEMENT - GRADIENT

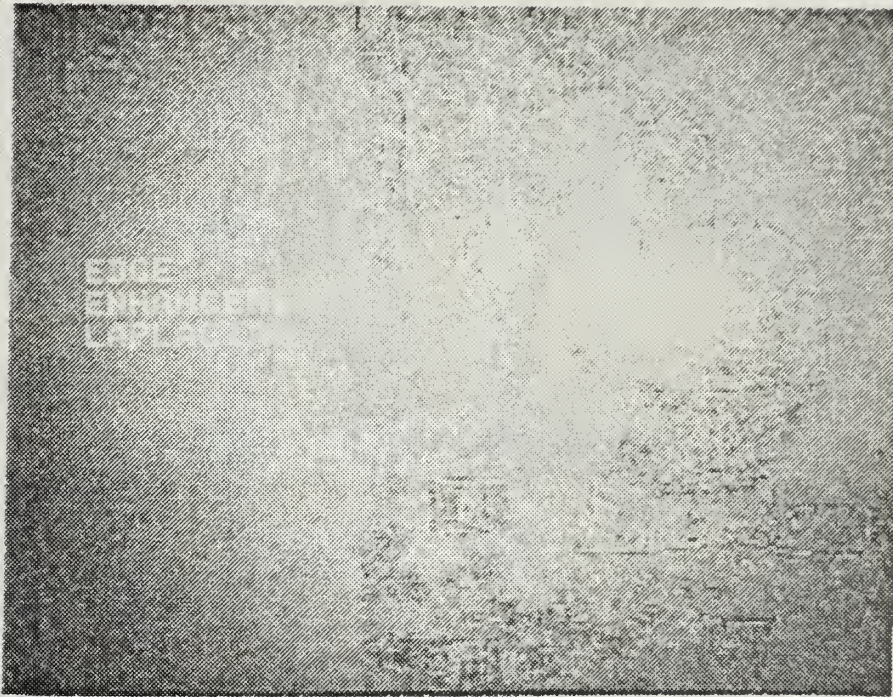


This picture brings out all the details as white lines against a black background using the gradient function for edge enhancement.





Figure 27. EDGE ENHANCEMENT - LAPLACIAN



The Laplacian method is used for edge enhancement for this picture. The Laplacian method emphasizes the maximum values within the image. Only these peak values are easily seen here.



Figure 28. LAPLACIAN Plus ORIGINAL



This picture is the result of adding the Laplacian picture to the original picture. There is an apparent increase in picture resolution for this image as compared to the original digitized picture.





## 11. Reducing Pictorial Noise

Noise, whether random or not, can appear in any image or picture as meaningless information concealing the true representation of the picture. This noise normally appears as spots (snow) in the picture. The types and sources of this noise can be based on where and how the original image was taken, equipment malfunctions, or problems with the method of digitization. An example could be the fluorescent screen of an x-ray inspection unit. Here the fluorescence may fluctuate or sparkle such that the digitized picture may be buried in snow.

There are two computer operations that can be used to alleviate the problem of noise. Spatial filtering can provide one effective way of noise filtering. In a low pass filtering operation of spatial filtering, the high spatial frequency interfering lines of a picture are reduced. This can be used on a noisy picture to reduce the high spatial frequency snow-like interference. In this operation a loss in picture resolution will result in the high spatial frequency components which were eliminated by the low pass filtering. This operation can be used when noise is not of a random nature.

The second method takes advantage of the randomness of the nature of the noise and at the same time does not sacrifice any resolution of the image. By adding together several digitizations of the same noisy image, the random noise can be reduced. The exact value of any pixel is dif-

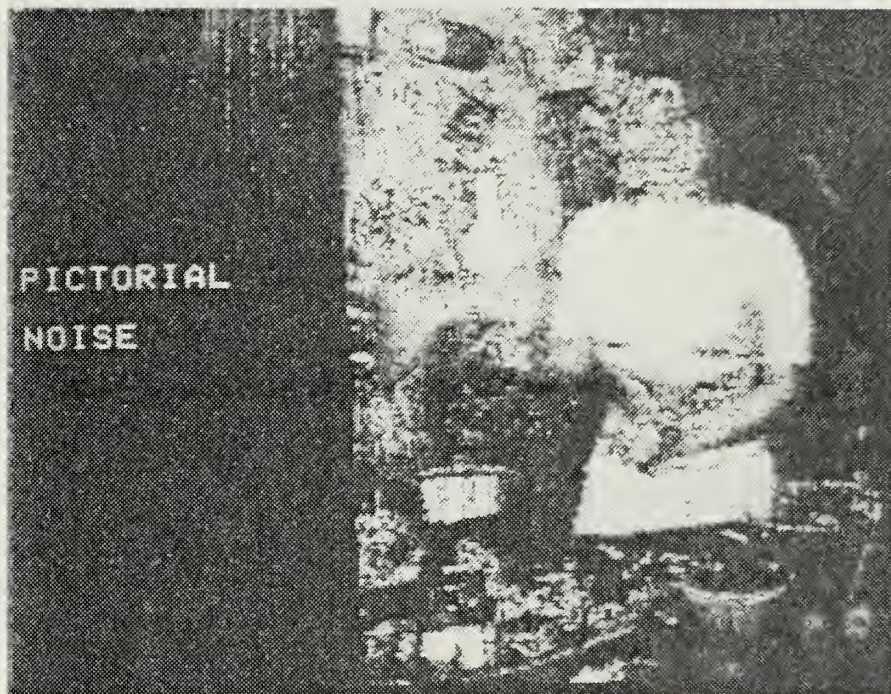




ferent with each digitization. This leads to the noise doing one of two things, either it adds to the nominal pixel value or it subtracts from it during digitization. The resulting average of this process over several digitizations will run to zero. The noise will be reduced by the square root of the number of pictures digitized. Figures 29 and 30 demonstrate this method of noise reduction.



Figure 29. PICTORIAL NOISE

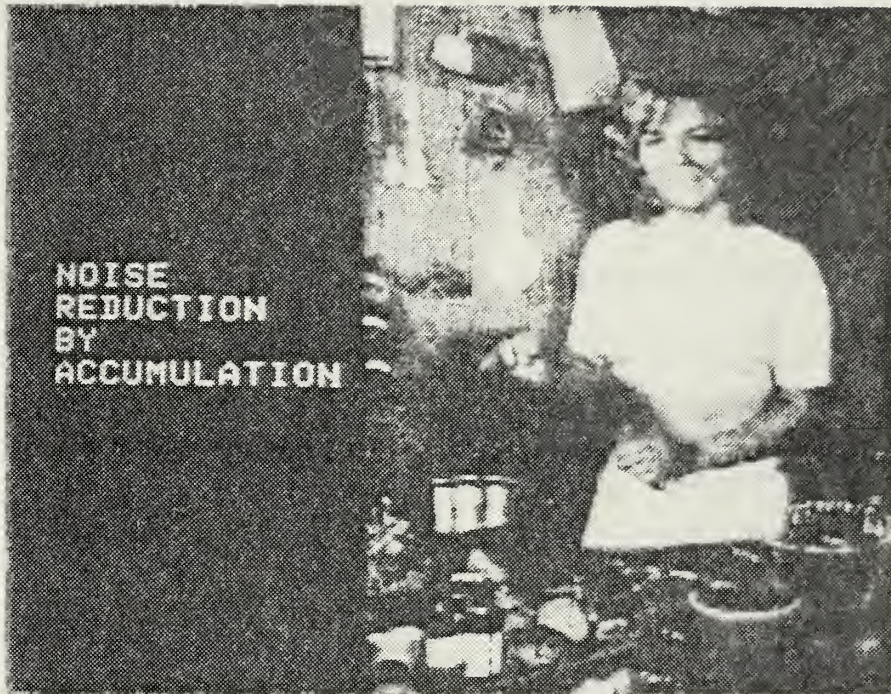


This is a picture showing the noise a picture may acquire during digitization.





Figure 30. NOISE REDUCTION By ACCUMULATION



The noise has been reduces considerably by adding together several digitizations of the same noisy picture. The averaging of these digitizations has reduced the amount of noise present when compared to the noisy picture in Figure 29.



## 12. Reducing Pictorial Data

There are 307,200 individual pixels that comprise a normal EyeCom picture. For each of these pixels, a shade of gray with a value between 0 and 255 exist. Each pixel requires eight bits of digital data to define the shade of gray. This amounts to more than two million bits of data to construct a complete EyeCom picture with full resolution.

With the characteristics of the human eye and the image, it is not always necessary that a full resolution picture exist for depicting objects and features of interest. The EyeCom picture allows for a greater picture resolution than the unaided human eye can resolve. This is related to the number of pixels used as well as the number of gray-levels. This makes it possible to reduce the picture resolution and associated data but still maintain all pictorial information that is necessary for image analysis. Reducing the number of gray-levels, as well as averaging the picture, can be performed without sufficient loss to resolution.

An averaged picture over an array of 2x2 of pixels and a reduction of number of gray-levels to 32 instead of 256 is a total reduction of one-sixth as many bits as the original. Instead of requiring two million bits of digital data, the same picture can be constructed using about 307,200 bits of data with little loss of resolution. Going further, an averaging of 4x4 array of pixels and a reduction to 16 gray-levels requires only 76,800 bits of data. At this stage, there may be some visual effect to the picture





such as slight blurring and loss of fine detail. Continued reduction of shades of gray, or further averaging will result in smaller data size, but will increase the loss of picture resolution.

In a practical situation, a determination must be made in the tradeoff between the type and degree of pictorial resolution loss and the amount of data required to handle the image by the system. The main advantage besides normal computing time and storage is in the transmission time. An original picture of two million bits would take about two hours to transmit over ordinary telephone lines, assuming a standard 300 baud rate for the telephone lines. A picture that has been reduced by an averaging of 4x4 array of pixels and only eight gray-levels can be sent in only three minutes. The amount of tradeoff can result in shorter transmitting time and smaller storage requirements as opposed to the amount of resolution lost in the picture due to the averaging and gray scale reduction.



## B. PSEUDOCOLOR OPERATIONS

Pseudocolor operates on the principle that each picture pixel has associated with it a unique brightness value or gray-level value. In normal operations, the EyeCom digitizes a image into a picture file containing 307,200 pixels, each with a gray-level value in the range of 0 (black) to 255 (white). By using the Image Processing programs for reducing the gray scale, the EyeCom reduces the gray-level range to any scale. Because the RAMTEK contains four memory planes (one memory plane each for red, green, blue, and white) and has the capability of loading a color lookup table with sixteen colors, the maximum range of pixel gray-level value displayed can only be from 0 to 15, or a total of 16 gray-levels instead of the 256 gray-levels displayed by the EyeCom [7]. Pseudocolor takes advantage of the fact that each of the 16 gray-levels can be assigned a unique color that is stored in the RAMTEK's Color Lookup Table.

To use Pseudocolor, a photograph or image is first digitized by the EyeCom into a 256 gray-level picture, and then the gray-level scale is reduced by a factor of sixteen to produce a picture with 16 gray levels. The 16 gray-level picture file is then transferred from RSX-11M on one PDP-11/50 to UNIX on another PDP-11/50. Because the picture (pixel) data is not in the correct format for display by the RAMTEK, i.e., EyeCom picture file has two pixels per word; whereas the RAMTEK picture file requires four pixels per word, the RSX-11M picture file requires packing. The



resulting UNIX picture file is then read into the MOS Refresh Memory of the RAMTEK and displayed on the screen at 240 lines of 640 pixels each [7].

The actual displayed picture (image) is formed on the RAMTEK's display screen in the following manner. As each pixel is displayed along the scan line, its gray-level value is associated with a corresponding color in the RAMTEK's Color Lookup Table. That color is then displayed on the screen for that pixel's unique gray-level. The resulting picture is a Pseudocolor representation of the original picture in that its appearance is similiar to a topographic map, i.e., the colors were arbitrarily chosen to represent a specific gray-level and may not be the true or natural color representation of that pixel. Hence, a Pseudocolor or 'False' color representation.

The purpose of Pseudocolor operations is the assignment of a particular unique color compared to the other fifteen colors to a pixel with the intent of improving the picture by amplifying certain areas for better picture analysis. In many applications Pseudocolor seems to enlarge human pattern recogniton capabilities.





## IV. CONCLUSIONS AND RECOMMENDATIONS

### A. CONCLUSIONS

The EyeCom Picture Digitizer and Display, a system which combined three types of man-made computer interaction - pictorial, graphic, and alphanumeric, did act as a single unit providing all the necessary operations required for efficient image processing. The EyeCom did perform the following image processing functions - magnification, averaging, gray-scale reduction, contrast enhancement, spatial filtering, edge enhancement, picture noise reduction, and histogram distribution.

The successful operation of these image processing functions were the result of the implementation of Drexel University's software conversion package of the original image processing programs used on the component RT-11 operating system of the EyeCom. The only modification to Drexel University's software package discovered during testing and implementation was related to the unique instruction set of the particular minicomputer used. Four instructions applicable only to PDP-11/34 minicomputer required recoding before successful implementation of the image processing pro-



grams on PDP-11/50 [5].

Once the EyeCom was implemented on RSX-11M operating system, preparation of a usable Operator's Manual was accomplished. The problems inherent in the development of this Operator's Manual were directly related to the user's level of knowledge of RSX-11M. Consequently, more than just a passing familiarity with the basics of RSX-11M was required. In particular, the concepts about assembling, compiling, task building, and executing a program and the I/O format of commands had to be mastered.

Since the purpose of this study was to operate the EyeCom Picture Digitizer and Display in a real-time environment, certain operations, i.e., averaging and edge enhancement, required many arithmetic calculations and took a couple of minutes to execute. As a result of these operations and picture digitization, the operation of the EyeCom in a real-time time-sharing environment was not realized. Any EyeCom image processing operation invariably tied-up the whole system, not only in time but also in memory space. Each individual picture file was extremely large (307K bytes) and when the EyeCom was operating, no other users could time-share. Consequently, the EyeCom can be best described as a "Single-User" system.

In order to use the Pseudocolor option, extremely large (307K bytes) picture files had to be transferred and processed to the RAMTEK Color Raster Scan Display. This processing required the packing of the picture file on another



minicomputer. The resulting picture was not a true color representation of the original digitized picture, but just a artificial representation of the 16 gray-level picture.

In conclusion, the EyeCom Picture Digitizer and Display performed adequately for black/white image processing and Pseudocolor operations as presently configured. However, the EyeCom was limited by the following: its black/white television and associated lack of a integral true color capability, its slow arithmetic calculations, its excessive use of memory space for storage of the individual digitized picture files, and its total use of a PDP-11/50 allowing for extremely limited time-sharing.

## B. RECOMMENDATIONS

Because the EyeCom , as presently configured, was limited by its black/white television camera, its slow arithmetic calculations used in image processing programs, and its dedicated use of the PDP-11/50, the following recommendations were made. First, the EyeCom software should be interfaced to the Macro Arithmetic Processor (MAP-300) to speed-up mathematical calculations and allow for better real-time operations. Second, if additional memory planes were added using the EyeCom Extension Chassis and if a color television camera were used, a self-contained color display





would be available, thereby eliminating the use of the RAM-TEK and another PDP-11/50. Third, using the available PDP-11/34 minicomputer would free the other PDP-11/50 and essentially provide a completely dedicated EyeCom system.

In the area of using the Macro Arithmetic Processor (MAP-300), the following information about its operations is provided. The Macro Arithmetic Processor (MAP-300) is an array processor which is capable of performing floating point calculations on large data arrays or data streams. The Macro Arithmetic Processor (MAP-300) operates as a peripheral device to a host computer system. The MAP-300 performs repetitive or reiterative operations requiring a large number of summations and multiplications. An example of this type would consist of correlations and convolutions, vector or matrix manipulations, fourier transforms, and data format conversion. The most significant area of importance for the MAP-300 is in operations where large batches or streams of data must be processed in real time and in floating point format [2] [3].

In general, the utilization of the MAP-300 would consist of reading the data into the memory of the MAP-300, perform the needed calculations, and read the data back into the host computer. The reading of the data would require configuring the MAP-300 buffer space, defining all constants, and writing the data to the buffer. The software that would performed the actual calculations would reside in the Simple Notation for Array Processing, Model II (SNAP-II), of the



MAP-300 [3]. This software would perform the needed calculations. Fortran level calls in the host computer would pass the necessary parameters to this area for the calculation. After the calculations were performed by the MAP-300, the results would be read back into the host computer. Programs that utilized a heavy amount of calculations would reside in the MAP-300. This would result in shorter execution times for these programs.

Concerning the color display, the EyeCom does have the capability, but that display requires the addition of eight more memory planes to provide three refresh memories, one for each of the primary colors -- red, green, and blue. Also, the black/white video scanner must be replaced with a color video scanner. These additional memory planes would produce 16 intensity levels for each of the three primary colors. In effect, there would be two shades associated with each of the 16 colors for display purposes. Presently, the RAMTEK only has the capability to display 16 different colors at any one time.

As far as using the PDP-11/34, the only problem associated with that recommendation is the purchase cost of the RSX-11M operating system. Once this RSX-11M operating system is available to the PDP-11/34 minicomputer, all the present image processing software could be easily switched over.



APPENDIX A  
OPERATOR'S MANUAL FOR THE  
EYECOM IMAGE PROCESSING SYSTEM  
USING RSX-11M OPERATING SYSTEM

SECTION	PAGE
1. INTRODUCTION .....	82
2. OPERATING PROCEDURES .....	83
3. USING IMAGE PROCESSING PROGRAMS .....	90
3.1 RUN COMMAND .....	90
3.2 PICTURE FILES .....	91
3.2.1 SINGLE FILES .....	91
3.2.2 TWO FILES .....	91
3.2.3 THREE FILES .....	92
3.3 PROGRAM PARAMETERS .....	94
3.4 RUNNING THE PROGRAMS .....	95
4. HOW TO PROCESS A PICTURE .....	96
5. PROGRAM DESCRIPTIONS .....	101
6. SUBROUTINE DESCRIPTIONS .....	128





## 1. INTRODUCTION

The Image Processing Programs described herein are for use with Spatial Data System's EyeCom Picture Digitizer and Display.

Operating procedures on how to use the EyeCom System are given in Section 2. Details on how to use the image processing programs are found in Section 3. Information on how to process a picture is provided in Section 4. Descriptions of the image processing programs are given in Section 5. Subroutine descriptions are provided for information only in Section 6. The image processing programs are designed to operate on a PDP-11/50 using the Real-time operating system RSX-11M.

The information contained in Sections 3,4,5, and 6 were obtained from Spatial Data System's Image Processing Programs for the PDP-11 [10]. Additional details about the programs and the subroutines in Sections 5 and 6 were obtained from Drexel university's Modification of RT-11 Image Processing Programs for RSX-11M [6].



## 2. OPERATING PROCEDURES

These operating procedures are for the USER of the EyeCom Picture Digitizer and Display. The USER can familiarize himself with the fundamentals of using RSX-11M and its basic programming principles by reading Digital Equipment Corporation's BEGINNER'S GUIDE [4] on RSX-11M found in the Computer Laboratory.

In the RSX-11M system, prompt characters '>' are typed by the computer to indicate that it is ready to receive typewritten instructions. A carriage return at the end of a typewritten instruction starts execution.

If during any processing of a picture file, an error occurs, e.g., "CSI ERROR", the picture automatically becomes "LOCKED" and is unusable. Consequently, the picture file that is locked must be unlocked as follows:

```
PIP Pj.PIX;l/UN <cr>
```



The following sections provide general instructions on how to operate the EyeCom Picture Digitizer and Display:

a. POWER-ON:

The EyeCom is turned-on by pulling out the "POWER" knob at the back of the display. The Vidicon television camera is turned-on by flicking the WHITE switch on the top of the camera ( RED light will come on ). The Light Table is turned-on by flicking the toggle switch on the left side of the camera mount. Additionally, the LA-30 Decwriter II must be on.

b. LOGIN:

The USER types the following:

HEL "USER NAME"

The SYSTEM responds with:

PASSWORD:

The USER responds by typing:

"USER'S PASSWORD"

The SYSTEM responds with a "WELCOME" message.

The USER is now ready to commence operations.

c. LINK TO THE EyeCom'S USER'S DIRECTORY:

To link to the operational space containing the EyeCom'S image processing programs, the USER types the following:

@[12,1]EyeCom

This command automatically establishes the EyeCom



USER's space in directory [12,1] and does the following:

1. Sets the LA-30 Decwriter II line length to 120 characters.
2. Sets the Directory as [12,1].
3. Sets the SYSTEM Main.
4. Installs the I/O Page.
5. Runs SRT108 startup program which clears the EyeCom picture display and turns off all the EyeCom keyboard buttons except Alphanumeric.
6. Deletes any picture files on [12,1].
7. Provides introductory remarks and information displayed on the LA-30 Decwriter terminal.
8. Asks if the USER wants to run the demo program that shows some of the image processing capabilities of the EyeCom.

d. INITIALIZATION PROCEDURE:

As stated in the Introductory Remarks, it is recommended that the USER run the "SETUP" program first to initialize the black and white levels.

The USER presses the "VIDEO" button on and "AN" button off on the EyeCom keyboard. Then the USER types the following in order pausing between each request to allow for SYSTEM queries:

RUN SETUP

The SYSTEM responds with the program prompt:

SET>





The USER types:

I

The SYSTEM responds with:

INITIALIZE ZERO AND RANGE

SET>

The USER types:

S S

The SYSTEM queries:

LENS CAPPED?

The USER responds by covering the lens and typing:

Y

The SYSTEM then iterates the Setup Register and responds with:

SET>

The USER types:

S W

The SYSTEM queries:

SET CURSOR ON WHITE

The USER responds by moving the Cursor with the Joystick to a white area of the picture and then pressing <ctrl E> to input this data.

The SYSTEM iterates the white range and responds with:

RANGE=xxx

SET>

The USER types:

S B



The SYSTEM queries:

```
SET CURSOR ON BLACK
```

The USER responds by moving the Cursor with the Joystick to a black area of the picture and then pressing <ctrl E> to input this data.

The SYSTEM iterates the black range and responds with:

```
ZERO=xxx
```

```
SET>
```

The USER has now completed the SETUP program and exits by pressing a <ctrl Z>.

e. EXAMPLE PROGRAM:

This example program creates a picture file as follows:

A picture or negative is placed on the Light Table. The "VIDEO" button on the EyeCom keyboard is ON. The image on the display screen is focused by adjusting the lens aperture.

The USER types:

```
RUN MOVE <cr>
```

The SYSTEM responds with the program prompt:

```
MOV>
```

At this time, the USER must specify a picture file or "TD" to move the video image displayed. The USER does this by typing:

```
SY:P0 = TV <cr>
```



The video image displayed is placed in an RSX output file called "P0.PIX;1". The "SY:" in the request refers to the system disc and is included in all responses requiring a picture file specification. "SY:" is NOT used before "TD" or "TV".

When execution is complete, the SYSTEM responds with:

```
MOV>
```

If the USER has no executions of "MOVE", the User exits by pressing a <ctrl Z> to terminate the "MOVE" program.

#### f. PSEUDOCOLOR OPERATIONS:

To display a RSX picture file (Pj.PIX;1) on the RAMTEK Color Raster Scan Display, the USER must first reduce the gray-scale of the picture file to 16 gray-levels, transfer the RSX picture file to UNIX, and then execute the RAMTEK Pseudocolor program.

To reduce the gray-level to sixteen, the USER must run the "SCALER" program.

To transfer the RSX picture file to UNIX, the USER types the following after logging on a UNIX terminal:

```
RSX /DPO/xx,xx/Pj.PIX/1 <cr>
```

When the file transfer is complete, i.e., UNIX responds with the prompt "%", the USER is now ready to display the picture file.

To display the RSX picture file on the RAMTEK, the USER types the following:

```
MV Pj.PIX PIC.RSX <cr>
```





EyeCom <cr>

At this time, a Pseudocolor picture will be displayed column by column on the RAMTEK Color Raster Scan Display.

g. SAVING USER PICTURE FILES:

When the USER is finished operating the EyeCom and he wants to save his created picture files (It is assumed that the USER has his own directory to store these picture files.), the USER can transfer these picture files by typing the following:

```
SET /UIC=[xx,xx] <cr>
```

```
PIP <cr>
```

The SYSTEM responds with:

```
PIP>
```

The USER responds with either:

```
=[12,1]*.PIX;* <cr> (MOVES ALL FILES)
```

OR

```
=[12,1]Pj.PIX;l <cr> (MOVES ONE FILE)
```

\*NOTE: If the USER fails to save picture files on his directory, the SYSTEM will destroy all picture files created on [12,1].

h. LOG-OFF:

To log-off, the USER types:

```
BYE <cr>
```



### 3. USING THE IMAGE PROCESSING PROGRAMS

The following notations are used throughout this USER'S Manual:

a) For file names,  $P_i$ ,  $P_j$ , and  $P_k$  represent any picture file or specifications. TV represents the Television Camera. TD represents the Picture Display.

b) `<cr>` is the carriage return key.

c) Lower case letters represent numerical values.

The Image Processing Programs operate on EyeCom digitized pictures. The pictures are stored on the computer disc memory under the names P0.PIX, P1.PIX, P2.PIX, etc. When the picture under the television camera is digitized, it appears to the program as though it were stored on the disc and is named TV. The pictures on the disc can be read from and written to the disc. Similarly, when a picture is read from device TV, the Scanner picture is digitized. To send the picture to or from the Picture Display, TD must be typed. All picture operations are performed by typing instructions on the computer terminal. The instructions for each program are divided into three parts: The command to start the program, the picture files to be used, and the program parameters.

#### 3.1 RUN COMMAND

The command to the computer to start executing the program consists of RUN followed by a space, the program name, and a carriage return. The carriage return '`<cr>`' initiates



the program.

Example: RUN PROFIL <cr>

Prior to the carriage return, errors may be corrected using the '<rub out>' key which erases individual characters, or the '<ctrl U>' key which erases the entire line.

### 3.2 PICTURE FILES

The list of picture files to be used by the program are typed below the RUN command. There are three cases: Single file, Two files, and Three files.

#### 3.2.1 SINGLE FILE

Programs that use only a single file require a single file name.

Example: RUN PHOTO <cr>

SY:Pj <cr>

Pj is used here to stand for any one of the file names, P0, P1, P2, TV, TD. "SY:" MUST be used before any Pj file, but never before TV or TD. "SY:" is shorthand notation for the system disc where the Task modules are stored.

#### 3.2.2 TWO FILES

Some programs use a picture file to be operated on (original picture) and a file for storing the result (resultant picture).



Example: RUN MOVE <cr>

SY:Pj = SY:Pk <cr>

Pj and Pk are used here to stand for any of the file names. In some programs the same file name can be used for both Pj and Pk. The following order must always be observed: RESULTANT FILE , EQUAL SIGN , ORIGINAL FILE. The program reads the original file and writes in the resultant file. The original picture file is not disturbed by the program, but the picture stored in the resultant file prior to running the program is erased and replaced with the resultant picture.

### 3.2.3 THREE FILES

Some programs use three files. There are two possibilities:

a) Two original pictures and one resultant. The two input files are separated by a comma:

Example: RUN COMBIN <cr>

SY:Pj = SY:Pk,SY:Pi <cr>

This program adds Pk and Pi and puts the result in Pj. Pj, Pk, and Pi can be any file name and not necessarily different:

Example: RUN COMBIN <cr>

SY:P0 = SY:P1,TV <cr>





This adds P1 to the digitized TV picture and writes the result in P0:

Example: RUN COMBIN <cr>

TD = SY:P2,TV <cr>

This adds the digitized TV picture to P2 and outputs the result to the Picture Display.

b) A program requires a double picture file to store a picture having more than 256 gray values. The files are then indicated in the same manner, i.e., separated by a comma:

Example: RUN SCALER <cr>

SY:Pj = SY:Pk,SY:Pi <cr>

Pk and Pi are two different files together containing one picture. Pj is the resultant picture.



### 3.3 PROGRAM PARAMETERS

The list of parameters instructs the program how to process the original picture. This list follows the picture file names with a slash preceding each parameter. The parameter is indicated by a letter followed by a colon (stands for equal) and the numeric value of the parameter:

```
Example: RUN CONTEX <cr>  
        SY:P0 = TV/B:b/W:w <cr>
```

The parameter names are capital B and W (standing in this case for black value and white value respectively). The lower case b and w stand for the numeric values of B and W. Thus, if it is desired to have the black value 50 and the white value 100, the instructions would appear as follows:

```
Example: RUN CONTEX <cr>  
        SY:P0 = TV/B:50/W:100 <cr>
```

Most programs have preassigned values for all parameters they use. If the parameters are not specified, these "default" values will be used by the program. Only those parameters whose values are desired to be different from the default values need to be typed.



### 3.4 RUNNING THE PROGRAMS

After typing the picture files and the parameters, the carriage return starts the program operating on the picture. Corrections can be made before the carriage return by using the '<rub out>' and '<ctrl U>' keys. If all the default values are desired, type the carriage return after typing the required file names.

If one changes his mind and does not wish to start the program operating on the picture, then '<ctrl Z>' will reset the computer to await a new program request.

The computer shows that it is ready for the USER to type in the RUN instructions by typing a right arrow cue ( '>' ). A program prompt (3 letters and '>') is typed by the computer when it is ready for the picture file names and parameters. After files and parameters are selected, the program will signal the USER that it has finished processing by issuing another program prompt for further input to that particular program.

```
Example: RUN CONTEX <cr>  
CON> SY:PO = TV <cr>  
CON>
```





#### 4. HOW TO PROCESS A PICTURE

A complete picture processor usually consists of many programs operating in sequence. Each program mathematically changes or combines pictures to produce a partial result. The original picture is usually obtained by digitizing the television camera image. Partial-result pictures are stored on the computer disc files. The final processed picture is outputted to the Picture Display. The disc files P0.PIX, P1.PIX, etc. are normally used as "scratch pads" to store intermediate results of the picture processing. There is no particular rule for choosing which file name is to be used in a program, but the following consideration must be taken into account.

If all files contain pictures that are no longer required, any convenient file or files may be used. Usually the picture processing starts by digitizing a picture using the instruction:

```
RUN MOVE <cr>
```

```
SY:P0 = TV <cr>
```

This places the digitized picture in P0. Of course, any other file can be used, but it is easier to remember that P0 is the file containing the original picture.

Once a file has been used to store the original picture or an important result, then the next program should place the result in a different file to avoid erasing that important result. Partial results are conveniently stored in as-



ending file numbers as the picture is processed with each program. All partial results are then available later if a different processing procedure is decided upon. However, in a long process all the files may be used before the final result is obtained, and some partial results may be lost.

For example, if the picture is to be contrast-enhanced and output to the Picture Display, then the following instructions could be used:

```
RUN MOVE <cr> (1)
```

```
SY:P0 = TV <cr>
```

```
RUN CONTEX <cr> (2)
```

```
SY:P1 = SY:P0 <cr>
```

```
RUN MOVE <cr> (3)
```

```
TD = SY:P1 <cr>
```

Instruction (1) digitizes the picture and writes it on file P0. Instruction (2) contrast-enhances the picture and writes the result on file P1. Instruction (3) outputs the result in file P1 to the Picture Display. The original picture is P0, the enhanced picture is P1, and they are both stored for use at a later time as possible original pictures for other programs. However, the picture from the television camera could be enhanced and recorded in one step by the following instruction:

```
RUN CONTEX <cr> (4)
```



TD = TV <cr>

The picture recorded on the Picture Display would be the same whether instructions (1), (2), and (3) are used, or just (4). The difference is that in (4) the original picture and result were not saved on the disc for future processing.

As an another example, consider the following series of instructions for removing shading and enhancing the contrast of a picture.

The first MOVE instruction digitizes the picture and the last MOVE records the result, P3, on the Picture Display.

The right-hand set accomplishes the same operation with only three instructions. The result of BOXCAR filtering is calculated directly from the digitized picture and placed in P0. The subtraction then takes place by digitizing the original picture again from the television camera, subtracting it from P0, and placing the result in P0. The contrast expansion of P0 is then performed and the result is recorded on the Picture Display. Thus, only the intermediate result after COMBIN is stored on the disc in file P0. Files P1, P2, and P3 are left undisturbed.

The right-hand example is an extreme case shown here to demonstrate the possible use of the picture files only. It has two drawbacks in actual practice. First, BOXCAR takes a few minutes to complete, and the television camera would have to scan the picture all the time the program is operat-



ing. It would be important to avoid moving the film or changing the light intensity during this time. Second, COMBIN program uses a second digitization of the picture which, due to the electrical noise in the television camera, is not exactly the same as the first digitization used in BOXCAR.

The left column shows one method and the right column another:

RUN MOVE <cr>

SY:P0 = TV <cr>

RUN BOXCAR <cr>

SY:P1 = SY:P0 <cr>

RUN COMBIN <cr>

SY:P2 = SY:P0,SY:P1/C <cr>

RUN CONTEX <cr>

SY:P3 = SY:P2 <cr>

RUN MOVE <cr>

TD = SY:P3 <cr>

RUN BOXCAR <cr>

SY:P0 = TV <cr>

RUN COMBIN <cr>

SY:P0 = TV,SY:P0/C <cr>

RUN CONTEX <cr>

TD = SY:P0 <cr>

The left-hand set of instructions stores the partial results on the disc as follows:

P0 - the original picture

P1 - result of BOXCAR filtering

P2 - result of subtracting the filtered





picture from the original

P3 - result of contrast expansion of the difference

A better method using only two picture files would be:

```
RUN MOVE <cr>
```

```
SY:P0 = TV <cr>
```

```
RUN BOXCAR <cr>
```

```
SY:P1 = SY:P0 <cr>
```

```
RUN COMBIN <cr>
```

```
SY:P1 = SY:P0,SY:P1/C <cr>
```

```
RUN CONTEX <cr>
```

```
TD = SY:P1 <cr>
```

The original and the result before contrast enhancement are stored in P0 and P1. If a second picture from the Scanner is to be processed in the same way, files P2 and P3 may be used in place of P0 and P1, so as to keep the first picture stored for future processing. Alternately, the same files may be used and the first picture erased.

When several programs are required to produce a result, it is recommended that an initial plan be formulated by writing down all the necessary partial results and assigning file names. Then those partial results that are considered not worth saving can be purposely erased, saving the important ones for later reference.



## 5. PROGRAM DESCRIPTIONS

All programs assume the picture to be stored in a contiguous file with the following format: 640 blocks of 512 8-bit bytes (256 16-bit words). Each block contains a single digitized column of data. The first block contains column 0, the left-hand column of the picture, and succeeding blocks contain columns digitized from left to right across the television screen.

The first 480 bytes contain the picture elements (pixels) digitized from the EyeCom and the remaining 32 bytes of each block are not used. Each pixel consists of eight bits of data (256 levels of gray). The pixels are ordered in top-to-bottom interlaced sequence with even-line pixels (0, 2, 4, ...) in even-addressed bytes and odd-line pixels (1, 3, 5, ...) in odd-addressed bytes.

The picture thus comprises a data matrix 640 pixels wide by 480 pixels high contained in a contiguous file of 640 blocks with each block 512 bytes long.

The following 24 programs can be executed (RUN) on the Spatial Data System's EyeCom 108PT using a PDP-11/50 and the RSX-11M Operating System:



RUN ACCUMT <cr>

SY:Pj,SY:Pk = SY:Pi/N:n <cr>

ACCUMT enables the accumulation of n multiple digitizations of a picture in a double-precision array (default n=1).

The pixel values stored in the single-precision file Pi are added to the double-precision pixel values stored in files Pj and Pk, for Pi=Pj or Pk. If Pi is TV, ACCUMT will add the Z data to the first run of ACCUMT to zero the double-precision files Pj and Pk. (These output files should be zeroed with GEN first before using ACCUMT.)

After the desired number of pictures are accumulated, SCALER may be used to reduce the double-precision file to the standard single-precision file.





RUN AVERAG <cr>

SY:Pj = SY:Pi/H:h/V:v <cr>

AVERAG divides the picture into elemental rectangles  $h$  pixels wide by  $v$  pixels high in file  $P_i$ , calculates the average value of pixels within each rectangle, and outputs to file  $P_j$  the average of the  $h$  by  $v$  blocks.

A lower-resolution picture is produced, where the pixels in each elemental  $h$  by  $v$  block have the average value of the original picture block. AVERAG reduces noise and graininess at the expense of geometric resolution.

Default values for  $h$  and  $v$  are 8.

RUN BALL <cr>

BALL generates a grid pattern in the EyeCom Graphics Display that matches the standard Spatial Data linearity or ball chart for adjustment of the Scanner linearity.



RUN BOXCAR <cr>

SY:Pj = SY:Pi/H:h/V:v <cr>

BOXCAR replaces each pixel in the original picture by the average of the surrounding pixels lying in the elemental rectangle by computing a sliding average on file Pi and placing the result in file Pj, j≠i. Each pixel in file Pj is the h by v average of the pixels about that pixel in file Pi.

Default values are h=32 and v=32.

RUN COLOR <cr>

COLOR renders a tri-color picture by requesting that the picture be digitized with a RED, GREEN, and BLUE filter respectively.

Recommended filters are:

RED WRATTEN 25

GREEN WRATTEN 47B

BLUE WRATTEN 58



RUN COMBIN <cr>

SY:Pk = SY:Pj,SY:Pi <cr> (ADDITION)

SY:Pk = SY:Pj,SY:Pi/C <cr> (SUBTRACTION)

SY:Pk = SY:Pj/C <cr> (COMPLEMENT)

SY:Pk = SY:Pj/C,SY:Pi/C <cr> (ADD COMPLEMENTS)

COMBIN adds (or subtracts) corresponding pixels in two pictures to produce a new picture which shows the sum (or difference) of the original pictures by adding the contents of file Pi to the contents of file Pj, dividing the sum by two, and storing the result in file Pk. If the switch C is present, the preceding file is complemented before adding. If Pi is not included in the command string, the contents of file Pj can be complemented by including the C switch.



RUN CONTEX <cr>

SY:Pj = SY:Pi/B:b/W:w/L:l/U:u <cr>

CONTEX increases the contrast of a picture by expanding the gray scale. CONTEX expands the contrast range of file Pi from the original span between B=b and W=w to a new span between L=l and U=u. Any values are acceptable from 0-255 for all arguments, b≠w. The modified picture is placed in file Pj, where:

$$(\text{new gray value}) = \frac{(\text{old gray value}) - B * (U - L) + L}{W - B}$$

The default values for L and U are 0 and 255, respectively. The default values for B and W are set to the 5% and 95% points, respectively, of the pixel intensity distribution. Any gray value between B and L (if B>L) will be set to L and any value between W and U (if W<U) will be set to U. All values from 0 to L or B, whichever is lower, and all values from 255 to U or W, whichever is higher, will be unaltered.

If U=L, all Z values between B and W will be set to U, and all others will remain unchanged. This allows the picture to be sliced into several gray contours.

Contrast enhancement brings out detail in low-contrast areas of the picture and is effective when used on the result of subtraction of two closely identical pictures using COMBIN.





RUN GEN <cr>

SY:Pi/S:s/H:h/V:v/X:x/Y:y <cr>

GEN generates a gray-scale test pattern stored in Pi of incrementing Z values in horizontal and vertical directions. The initial, upper left pixel value is S. Pixel values are incremented by the value of h every x pixels horizontally, and by the value of v every y pixels vertically. Default values for s, h, and v are zero, and for x and y are one.

1. Example:

RUN GEN <cr>

SY:P1/H:1/V:1 <cr>

Produces a continuous gray-scale wedge in P1 consisting of all pixel values from 0 to 255 repeated approximately two times horizontally and vertically.

2. Example:

Run GEN <cr>

SY:P1 <cr>

Produces a picture in P1 with all values equal to zero (all black picture). Used to zero files prior to using ACCUMT.



RUN GRADE1 <cr>

SY:Pj = SY:Pi <cr>

GRADE1 approximates the magnitude of the gradient of a full picture, i.e., the square root of the sum of the squares of the horizontal and vertical derivatives at each point of the picture as follows:

$$G(x,y) = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2}$$

by replacing each pixel  $P(i,j)$  with the following:

$$\left[ \sum_{k=i-1}^{i+1} P(k,j+1) - \sum_{k=i-1}^{i+1} P(k,j-1) \right]^2 + \left[ \sum_{k=j-1}^{j+1} P(i+1,k) - \sum_{k=j-1}^{j+1} P(i-1,k) \right]^2$$



RUN GRADE2 <cr>

SY:Pj = SY:Pi <cr>

GRADE2 approximates the magnitude of the gradient,

$$G(x,y) = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2}$$

by replacing each pixel  $P(i,j)$  with the following:

$$\left| \sum_{k=i-1}^{i+1} P(k,j+1) - \sum_{k=i-1}^{i+1} P(k,j-1) \right| + \left| \sum_{k=j-1}^{j+1} P(i+1,k) - \sum_{k=j-1}^{j+1} P(i-1,k) \right|$$

Calculation time for this approximation is substantially shorter than for that of GRADE1.





RUN HISTOR <cr>

SY:Pi/W <cr>

HISTOR calculates a histogram or frequency distribution of gray-scale values of a picture. The histogram is shown on the Picture Display. Any gray-scale value may be selected by the Joystick Cursor, and the exact number of pixels of that value printed. The gray value of a point on the picture selected by the Joystick Cursor can also be printed. Up to 10 gray-scale values may be selected from the histogram display and stored for subsequent partition of the gray scale using CONTEX.

HISTOR is a collection of routines based on the histogram statistic of a digitized picture Pi (or a subwindow, by use of the W switch). (If the W switch is used, a beep is sounded twice for the window input with the Joystick Cursor. Press '<ctrl E>' while the cursor is at two diagonally opposite corners of the desired portion of the picture.)

The program reads the picture (or window) and generates a scaled histogram in the EyeCom graphics overlay section. The histogram represents the picture's gray-scale distribution function, from black=0 at the left to white=255 at the right.

After typing '<H> <cr>', a cue (\*) is printed and the program waits for any of the following function keys to be typed:



KEY	FUNCTION
<+>	Increase the amplitude of displayed histogram.
<->	Decrease the amplitude of displayed histogram.
<G>	Sample one value on the GRAPH (indexed by the Joystick) and print amplitude and gray value.
<C>	As in <G>, but enter the gray value in a PARTITION Table and tick-mark the histogram baseline.
<X>	The program waits for an output file Pj after building a lookup table from partition set. When a picture has been contoured by <X>, a new output file, Pj, is loaded with the gray values from the created PARTITION Table.
<E>	Erase points from baseline.
<R>	Redraw the current histogram.
<P>	Sample a point in the input picture file, indexed by the Joystick, and print the (X,Y) coordinates and the gray-scale value (Z) at that point.

Typing a '<ctrl Z>' causes an exit from HISTOR.



RUN IRIS <cr>

IRIS is a demonstration program that loads the EyeCom Color Lookup Table with 32 colors and continuously circulates this color assignment.

The 32 colors are assigned to the most significant bits of the Picture Display data.

This program is to be used with a COLOR TV CAMERA.

#### IRIS COLOR ASSIGNMENTS

Gray Scale Value	Color Name	Primary Color Brightness Value		
		R	G	B
31	White	255	255	255
30		192	192	255
29	Light Blue	128	128	255
28		64	64	255
27	Blue	0	0	255
26		0	64	255
25	Blue Cyan	0	128	255
24		0	192	255
23	Cyan	0	255	255
22		0	255	192
21	Cyan-Green	0	255	128
20		0	255	64
19	Green	0	255	0
18		64	255	0



17	Yellow-Green	128	255	0
16		192	255	0
15	Yellow	255	255	0
14		255	192	0
13	Orange	255	128	0
12		255	64	0
11	Red	255	0	0
10		255	64	64
9	Pink	255	128	128
8		255	64	128
7	Magenta-Pink	255	0	128
6		255	0	192
5	Magenta	255	0	255
4		192	0	255
3	Violet	128	0	255
2		128	64	192
1	Gray	128	128	128
0	Black	0	0	0

Gray Scale - Decimal equivalent of five most significant bits.

Color Name - Without names are intermediate colors.

Brightness Value - Decimal equivalent of 8-bit brightness values.





RUN LABEL <cr>

SY:Pj/GL:n/GR/TD <cr>

LABEL draws characters and character strings on any digitized picture file, Pj.

GL:n allows the USER to specify the gray level of characters ( $1 \leq n \leq 255$ ) if the graphics mode is not used. A gray level of 0 is interpreted as GL:2. In the graphics mode, only the least significant bit (LSB) of n is important.

If LSB is one ( $n=1$ ), the graphics bit is set.

If LSB is zero ( $n=0$ ), the graphics bit is cleared.

GR implies the graphics mode characters. In this mode only the graphics bit of the picture will be altered.

TD forces parallel output to the Picture Display to allow visual inspection. (Not Applicable if output is TD.)

Defaults:

/GL:n	n=255	in PIX mode
/GR	/-GR	PIX mode
/TD	/-TD	No parallel TD output

LABEL will prompt for horizontal or vertical lettering (H or V).

Position the Joystick Cursor to enter the coordinates of the first character followed by a '<ctrl E>'.



Type characters PAUSING momentarily after each character to allow for computer processing. Last character typed can be removed by pressing the '<rubout>' key.

Pressing '<cr>' during character input will move the input to the next line for horizontal lettering or to the next column for vertical lettering.

Terminate character input by pressing '<ESC>'.

To reposition the area for lettering, terminate your present position with a '<ESC>' and respond with a 'Y' '<cr>' or 'N' '<cr>' to the query "ANY MORE CHARACTERS?".



RUN LAPLAC <cr>

SY:Pj = SY:Pi <cr>

LAPLAC calculates the Laplacian, i.e., the sum of the horizontal and vertical second derivatives at each point of the picture.

LAPLAC approximates the Laplacian as follows:

$$L(x,y) = \left( \frac{\partial z^2}{\partial x^2} \right) + \left( \frac{\partial z^2}{\partial y^2} \right)$$

by replacing in file Pj the value of each pixel in file Pi minus the average of the eight surrounding adjacent pixels. Zero results are represented as middle gray, Z=128.





RUN MFILT <cr>

OUTPIX/H,OUTTMP = INPIX/X:x/Y:y/IX:ix/IY:iy/AV:av/G/W,TMPPIX

MFILT convolves a selected region with a picture.

OUTPIX is the output file/device the filtered picture is to be output to. If the /H (Half Resolution) switch is invoked, pixels will be duplicated in the Y direction; however, only every other line will be output leaving alternate lines untouched. This switch speeds compilation and is useful if the output is to a file for subsequent display on TD.

NOTE: If OUTPIX is TD, the /H switch is forced by the program. OUTPIX is optional if USER desires to record templates to OUTTMP.

INPIX is the input device/file to be filtered.

X is the X size of the template (Default=20).

Y is the Y size of the template (Default=20).

IX is the X coarseness of the filter. If IX=1, all points in the X direction of the template will be used. If IX=2, every other point in the X direction will be used, etc. (Default=1).

IY is the Y coarseness (Default=1).

AV is the number of times the template will be averaged



from the template input picture (Default=1).

G invokes the general filter. In this case, the template can be input from the keyboard or a template file.

W invokes a window on the input and output pictures. If this is invoked, two points are entered via the Cursor ('<ctrl E>') which define the diagonal of the rectangular window. Only points within the window are filtered.

TMPPPIX is the file/device the template will be input from if the 'G' switch is not invoked. If this optional file is not specified, the default file/device will be INPIX. An area X by Y will be displayed around the Cursor. Enter '<ctrl E>' AV times to enter and average the input template.

OUTTMP is an optional RSX-11M output file specification. If this field is specified, the template used (whether from INPIX, TMPPPIX or via the 'G' switch) is written to this file. There is no default file name for this field.

The 'G' Switch:

This switch allows the template to be entered numerically from the keyboard or to retrieve a template from a previously recorded template file. Invoking this switch will cause a prompt of:

"INPUT FROM THE KEYBOARD?"

If this answered with 'Y', the program will prompt for a



template X by Y with increments of IX and IY. Template values must be 0 and  $\leq 255$ .

If the answer is other than 'Y', it is assumed that the template will come from previously recorded template file. The program then prompts for this file. If the X, Y, IX, or IY values of the template file do not match the values of the first input string, a warning is issued and the values which the template was recorded with are used.

Invoking the 'G' switch cancels TMPPIX and the AV switch of the first string.

RUN MOVE <cr>

SY:Pj = SY:Pi <cr>

MOVE transfers a copy of picture file Pi to file Pj. If Pi=TV, a full picture is digitized from the television camera and stored in file Pj. If Pj=TD, the picture file Pi is output to the Picture Display.



RUN OHART <cr>

OHART maps the particular color to octal code on either a white or a black background.

RUN PHOTO <cr>

SY:Pi/X:x/Y:y/H:h/V:v/L:l/W:w <cr>

PHOTO prints a low-resolution facsimile of the full EyeCom picture by producing a printed image of the picture file Pi. The (x,y) are the coordinates of the upper left corner of the picture, h and v are the horizontal and vertical increments, and l and w are the horizontal and vertical range of the picture. The default printing format is a full page-width square picture with selected typewriter characters to produce a "gray scale". Resolution is 80x50 characters (10"x10") with 22 Z values corresponding to 22 ASCII characters used to represent the gray scale.

Default values are x=6, y=0, h=6, v=0, l=50, and w=80 producing a picture size 80 by 50 characters, suitable for the LA-30 Decwriter.





```
RUN PROFIL <cr>
```

```
SY:Pi/X:x/Y:y/L:l/B:b/A:a/D <cr>
```

PROFIL prints the Z-values of a vertical row of pixels. It reads from file Pi a band 'b' lines wide with line 'x' being the center line number. Each line is read 'a' times before PROFIL averages this data into a single line profile. A vertical line is read. The starting pixel coordinate is given by 'y', and 'l' pixels in each line will be read. The resultant averaged profile will be printed, and a full-height profile displayed in the graphics overlay.

If /D is included in the command, PROFIL will respond by typing "CALIBRATE" on the terminal. The program instructs the USER to make a calibration from the television camera. After the calibration is completed, PROFIL will type a profile in density units as specified in the command line.

NOTE: The program must be calibrated the FIRST TIME the 'D' switch is specified or a FORTRAN error will result. The calibration factors will be retained through-out the duration of the program or until changed.

```
Example: RUN PROFIL <cr>
```

```
TV/A:3/X:256/Y:0/L:10 <cr>
```

This example types a profile of pixels 0-10 in column 256 digitized from the television camera. Each pixel is digitized 3 times and averaged.



Default values are  $x=Jx$ ,  $y=Jy$ ,  $l=Lmax$ ,  $b=1$ , and  $a=1$  where  $Jx$  and  $Jy$  are coordinates of the Joystick Cursor and  $Lmax$  is the number of pixels between  $y$  and the bottom of the picture, i.e.,  $Lmax=480-y$ . Press '<ctrl E>' to input the Joystick coordinates.

See WINDOW for an example of calibration procedure when using the 'D' switch.

RUN SCALER <cr>

SY:Pk = SY:Pi,SY:Pj/D:d <cr>

SCALER reduces a double-precision picture in files  $P_i$  and  $P_j$  to a single-precision picture in file  $P_k$  by dividing each pixel by  $d$ .

This program is primarily used with ACCUMT to produce a single-precision file for output to the Picture Display.

The default value is  $d=1$ .



RUN SETUP <cr>

SETUP is used to adjust the EyeCom Setup, Zero, and Range Registers prior to digitizing the Scanner Picture. It also enables or disables display functions under the keyboard control. The program prompts with "SET>". The following commands are then initiated from the keyboard.

A message is printed after each key is typed indicating the command selected.

<I>: Initializes the zero (0) and range (255) registers, enables the television Picture, and disables the Picture and Graphics Displays.

<S>: Setup (adjust) the EyeCom Setup, Range, and Zero Registers. Follow '<S>' with one of the following:

<S>: Setup Register. Instructs the USER to cap the camera lens and then repeatedly digitizes a coarse grid and adjusts the setup register to provide an average zero black level.

<B>/<W>: Black/White Exposure. Prompts the USER to indicate black and white points on the picture. Uses the digitized values



to calculate new values for the Zero and Range Registers. New values are loaded into the registers and cause the Z values corresponding to the selected black and white points to be digitized as 0 and 255.

<P>: Profile. Generates a real-time vertical density profile of the camera input at the column indexed by the Joystick Cursor. It prompts with "INPIX>". Response is 'TD'. By repositioning the Joystick Cursor and pressing '<ctrl E>', the vertical density profile line will move to the Cursor's new position. Terminate with a '<ctrl Z>'.

<C>: Clears (erases) the contents of the display selected by typing one of the following:

<A>: Alphanumeric Display

<G>: Graphics Display

<P>: Picture Display

<F>: Fliker. Alternates between television picture and Picture Display at television frame rate. Terminate with a '<ctrl Z>'.





<D>/<E>: Disable or Enable display selected by  
typing one of the following:

- <A>: Alphanumeric Display
- <B>: Black/White on Color Display
- <C>: Joystick Cursor Display
- <G>: Graphics (one-bit) Display
- <L>: Logarithmic Mode
- <M>: Graphics Mode
- <O>: Overlay Color Display
- <P>: Picture Display
- <S>: Video Scanner (Camera)
- <Z>: Psuedocolor Display on Color Monitor

RUN SHIFT <cr>

SY:Pj = SY:Pi/L:l/R:r/U:u/D:d <cr>

SHIFT moves the picture in file Pi left, right, up, or down the number of pixels specified in l, r, u, or d, or any combination by storing the result in file Pj and zeroing the empty areas.

Default values are l=0, r=0, u=0, and d=0.



RUN WINDOW <cr>

SY:Pi/X:x/Y:y/H:h/V:v/D <cr>

WINDOW reads a 20 by 20 array of pixels and types the gray values or the density values on the terminal. The printed pixel values may represent adjacent pixels or pixels spaced farther apart, i.e., every other pixel, every 3rd pixel, etc. The pixel spacing is selected independently in the horizontal and vertical directions. To type the density values, the 'D' switch must be included. WINDOW will request two calibration values. The upper left corner of the array starts at (x,y) and h and v are the horizontal and vertical increment values.

Default values are  $x=J_x$ ,  $y=J_y$ ,  $h=1$ , and  $v=1$  where  $J_x$  and  $J_y$  are the coordinates of the Joystick Cursor. Input the Joystick coordinates by pressing '<ctrl E>'.

The calibration procedure when using the 'D' switch is shown in the following example:

CALIBRATE

TYPE DENSITY: 1.12 <cr>

SET CURSOR: set Joystick to 1.12 standard  
press '<ctrl E>'

(120,131)=101: x,y coordinates and Z value

The computer types profile or window values in density units.



RUN ZOOM <cr>

SY:Pj = SY:Pi/X:x/Y:y/H:h/V:v <cr>

ZOOM magnifies a portion of the picture by expanding each pixel in the original picture file Pi by integral factors of h in the horizontal and v in the vertical direction into h by v pixel blocks. The desired portion of the picture is indicated by the Joystick Cursor. The subframe to be expanded has its upper left-hand corner at (x,y) in the original picture. The expanded picture will be stored in picture file Pj.

Default values are x=128, y=120, h=2, and v=2.



## 6. SUBROUTINE DESCRIPTIONS

This section on Subroutines is included for information purposes only. Subroutines can not be executed independently as Programs. Only the Programs in Section 5 which call Subroutines can be executed (RUN).

The specific details and subroutine listings can be found in the EyeCom Image Processing RSX-11M Documentation.

The following is a brief description of the Image Processing Subroutines:

ASC RAD - converts a string of ASCII characters to radix 50 packed form.

ASYNIO - keyboard input/output routine.

BINDEC - converts binary to decimal ASCII characters.

BINOCT - converts binary to octal ASCII characters.

CALIBR - accepts two sample density values at keyboard and returns slope and intercept values for linear approximation of the optical density scale.

CHRTAB - a character table.

C MAP - maps each of the 32 gray levels into a combination of color gun intensities.





CORLAT - sums the product of the corresponding points of a picture within a specified rectangle.

CSI - requests a command string from the keyboard in RSX-11M standard format, passes control to PARSE to check the validity of the command and to build an internal table to open files, and then initializes and opens each file by assigning a channel number in sequence.

CURSOR - reads the Joystick Cursor coordinates.

COORD - passes calling coordinates to the program.

CVIDIG - points digitized data to the output screen.

DECBIN - converts decimal ASCII characters to binary.

DISPLAY - controls the Display register from the EyeCom keyboard.

DKIO - reads and writes data from a specified device.

ERASE - clears the Alphanumeric, Graphic, or Picture Display on the EyeCom keyboard.

GETDEV - lookups and returns the currently assigned channel number of the device.

GETPUT - gets or puts a single pixel to or from the picture display memory.



GRAPH - generates a bar or point graph with baseline  
in the EyeCom graphics refresh memory.

GRYSCL - outputs a gray scale "width" pixels wide to  
the channel specified.

HISTOG - builds a double-precision histogram table  
from a specified rectangular picture window.

INTR45 - linearly interpolates a fifth line of  
picture from four lines of input and  
places the output in a specified buffer.

IOPAG - establishes input/output page which is common  
to all programs and defines the following  
registers: Command/Status, Display, X, Y, Z,  
Setup, Zero, Range, Keyboard Data, Gray  
Value, Alphanumeric Status, Alphanumeric  
X and Y Address, Display Data, Red, Green,  
Blue, Clock, and Program Status.

KBIO - EyeCom keyboard routine.

LINE - draws a vector in the graphics display.

MACROS - emulates PDP-11 hardware instructions.

PARSE - processes an ASCII line filling a table with  
the device and file specification and assigns  
the values of switches present.



PIXCHR - outputs picture characters to channel.

PRODIG - reads the file on the channel and calculates a vertical profile averaged over many full or partial columns.

SAVRES - saves and restores registers.

SCREEN - returns the maximum dimensions of the screen and the visible column length.

SELECT - accepts a keyboard character and jumps to the location specified in a address table.

SETSZR - setups zero and range register.

SUBST - substitutes 256 new values for Z values.

SWTSET - scans all dataset specifiers in the table.

SYMBOLS - hardware register definition for EyeCom.

TDIN - gets data from the terminal display.

TDIO - reads/writes from the picture display memory.

TDOUT - writes data to EyeCom Picture Display.

TD\_SCL - outputs a gray scale "width" pixels wide to the picture display.

TVIO - digitizes a window from the High-Resolution Digitizer and outputs to the Picture Display.



APPENDIX B

BASIC OPERATING PRINCIPLES of the EyeCom

SECTION	PAGE
1. INTRODUCTION .....	134
2. SCANNER AND PICTURE DISPLAY .....	136
2.1 Scanning Accuracy .....	136
2.2 Photometric Accuracy .....	137
2.3 Lens .....	138
2.4 Resolution .....	138
2.5 Linearity .....	139
2.6 Lag .....	139
3. OPERATING MODES .....	140
3.1 Linear Mode .....	140
3.2 Log Mode .....	140
3.3 Photographic Density .....	140
4. SYNC GENERATOR .....	142
5. SCANNER AMPLIFIER CHAIN .....	142
6. DIGITIZERS .....	143





7.	GRAPHICS AND PICTURE DISPLAY .....	144
8.	MEMORY PLANE ORGANIZATION .....	147
9.	ALPHANUMERIC DISPLAY .....	148
10.	INTERFACE CONTROLLER .....	148
11.	JOYSTICK .....	149
12.	CONTROL AND DATA REGISTERS .....	151
12.1	Command/Status Register .....	151
12.2	Display Register .....	152
12.3	X, Y, and Z Registers .....	152
12.4	Setup, Zero, and Range Registers .....	152
12.5	Keyboard Status and Data Registers .....	152
12.6	Alphanumeric Registers .....	152
12.7	Graphics Data Register .....	153
13.	CALIBRATION .....	153
14.	DIGITIZING THE PICTURE .....	156
14.1	Normal Mode .....	156
14.2	Sample and Increment Mode .....	157
14.3	Digitizer Cursor .....	158



## 1. INTRODUCTION

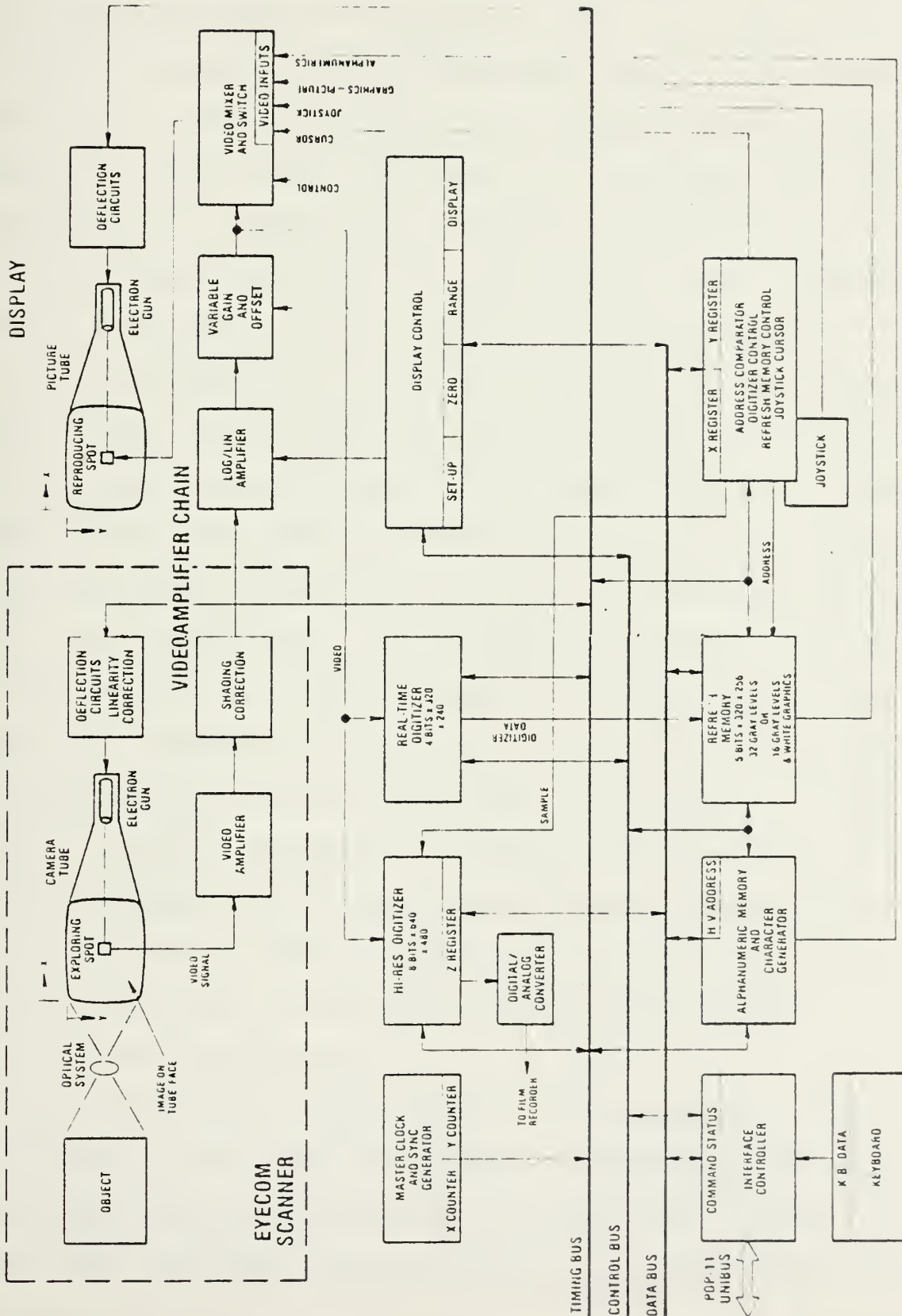
All the material contained in Appendix B is referenced from Spatial Data System's EyeCom Handbook [9].

The EyeCom scans and reproduces pictures using a scanning system that is compatible with standard television. This system forms the operating basis of the displays and digitizers. The 'Scanner' shown and addressed in the following sections is normally part of the original equipment issued with the EyeCom by the manufacturer, Spatial Data. The EyeCom System in the Computer Laboratory does not have this original Scanner but has the Vidicon television camera instead. Since this Appendix details Spatial Data's hardware, the description of the basic operating principles of the EyeCom is for the original equipment.

The major functional blocks of the EyeCom are shown in the System Diagram, Figure 1.



Figure 1. EyeCom SYSTEM DIAGRAM





## 2. SCANNER AND PICTURE DISPLAY

The Vidicon television camera tube senses the light emitted by the object to be digitized. A light image of the object is produced on the photosensitive face of the camera tube by an optical system containing an ordinary camera lens. In normal use, the image is stationary and the brightness at any point on the image is a function of an X and Y coordinate position on the tube face. Image brightness at that point is then defined as a third dimension, Z.

The brightness, or Z-value, is converted into an electrical voltage (video signal) by repeatedly scanning the image with an exploring spot formed by the electron beam of the camera tube. This spot generates an electrical video signal which indicates the brightness at its instantaneous position. The video signal is transmitted through a video amplifier chain to the display, where it controls the brightness of a reproducing spot formed by the CRT electron beam. The reproducing spot moves over the viewing screen in a path similar to and synchronous with the exploring spot. The reproducing spot reconstructs the brightness distribution in the image area, both in magnitude (Z) and position (X,Y), and thus generates a television picture of the object.

### 2.1. Scanning Accuracy

The exploring spot formed by the electron gun scans the image on the camera tube face. Consequently, the size of the spot remains constant as it scans the image. In fact, the size does change slightly as it moves toward the





edges of the picture. Since the spot size has a direct bearing on the scanner resolution, the EyeCom Scanner electron optics are designed to minimize changes in spot size to produce a more uniform resolution over the entire image.

The exploring spot was also assumed to move at a perfectly constant rate producing a linear scanning raster. In fact, the electron optics introduces significant variations that produce a nonlinear scan. Special compensating circuits in the EyeCom Scanner correct the scanning rate to provide a linearity over four times better than normal television cameras.

## 2.2. Photometric Accuracy

The ideal scanner produces a video signal that accurately represents the light values in any part of the image. Practical scanners are limited by the total range of light values that can be sensed in the image. That is, the ratio of maximum to minimum light value has a practical limit referred to as the dynamic range. The dynamic range for the vidicon camera tube is 0.65, thereby giving the vidicon tube a range ratio of over 200:1. When the Scanner is used to scan photographic film, this value corresponds to an equivalent contrast range of 2.2D. The digitized values are also linearly related to light values measured on the density scale over the dynamic range. This is referred to as the photometric linearity of the scanner. The video amplifier of the EyeCom Scanner is designed to optimize resolution and photometric linearity.



Another aspect of photometric accuracy is uniformity of response to light values over the entire image. For a constant illumination over the entire image, the scanner response should be the same. This should be true both for the darkest and lightest usable illumination. The variation in response in these two cases is corrected by special circuits in the EyeCom Scanner.

### 2.3. Lens

The EyeCom Scanner is supplied with a standard C-mount 25-mm f/1.4 television camera lens that has sufficient magnification for usual applications. The magnification of the image as seen on the Picture Display is the product of the optical magnification of the lens system and the electronic magnification of the camera tube image. The display screen is approximately 10 inches (25 cm) wide. Thus, the 13-mm wide image is magnified electronically by a factor of 20.

### 2.4. Resolution

The EyeCom samples 640 points across the image. Thus, the finest structure that can be represented by the digital data is 640 alternate bright and dark samples. An image containing finer structure may produce spurious patterns in the digital pixel data. However, the resolution of the EyeCom Scanner is purposely limited to avoid spurious patterns, thus allowing the system to produce digital data which accurately represents the image.



The resolution of the Scanner stays within small limits over the entire image. Typically, the resolution decreases or the spot size increases in the corners of the image to provide a camera resolution of 50% MTF (Modulation Transfer Function) of greater than 450 lines, only about 10% less than the central resolution of 500 lines. Relating resolution to spot size for the camera image, 13-mm wide, the calculated spot size is approximately 30 microns at the edges and 25 microns at the center. The spot therefore changes size by 5 microns or 0.05% of the image height (10-mm).

#### 2.5. Linearity

Because the path of the scanning spot differs from the optimum by introducing a slight distortion in the picture, the Scanner deflection system is designed so that the array of pixels can be adjusted to lie close to their ideal position, a rectangular-element grid 640 pixels wide by 480 high. The maximum error amounts to less than  $\pm 1/2\%$  of the image height for all pixels.

The linearity is achieved by geometric correction waveforms. Adjustments are provided for horizontal, vertical, and corner linearity as well as correction for perpendicularity of the horizontal and vertical deflections.

#### 2.5. Lag

When the image on the EyeCom Scanner is suddenly changed or moved, there is a finite time delay for the camera tube to accommodate to the new conditions. This lag has



an effect on the operation only when the image is in motion. Rapidly moving objects leave a trailing image which may interfere with the analysis in some applications.

### 3. OPERATING MODES

#### 3.1. Linear Mode

When the EyeCom is used in the linear mode, the video signal is digitized into 256 equal steps. With the vidicon tube, the digital output value (Z) is expressed as:

$$Z = K * I^{\gamma} \quad (1)$$

where I is the image brightness and K is a proportionality constant.

#### 3.2. Log Mode

When the EyeCom is used in the logarithmic mode, the logarithm of the video signal is digitized in 256 equal steps. These steps correspond to equal logarithmic intervals (light ratios). The digital output is given by the logarithm of equation (1):

$$Nd \approx \log Z = \log K + \gamma \log I \quad (2)$$

Thus, gamma becomes a constant scale factor in the transfer function and the digital output is proportional to log I.

#### 3.3. Photographic Density





When the EyeCom is used to scan a photographic transparency illuminated by the optional Model 700LT Light Table, the light from the Light Table is attenuated by the transmittance of the film. Transmittance (T) is expressed on a scale from zero (completely opaque) to one (completely transparent) and is given by the ratio of the light coming from the film (I) to the incident light from the Light Table (I<sub>o</sub>) as follows:

$$T = \frac{I}{I_o} \quad (3)$$

The photographic density of a film is commonly used to measure the transmittance and is defined as:

$$D = \log \frac{1}{T} \quad (4)$$

The relationship between light from the photograph and photographic density is found by substituting (3) into (4):

$$D = \log \frac{I_o}{I} = \log I_o - \log I \quad (5)$$

Equations (5) and (2) show in the logarithmic mode that the digital value is directly proportional to  $\log I$  and, therefore, is inversely proportional to photographic density (D).



Using the Model 700LT Light Table with black and white transparencies, the equivalent sensitivity of the standard vidicon tube is approximately ASA 200 and an equivalent contrast range of 2.2D.

#### 4. SYNC GENERATOR

The signals for synchronizing the exploring spot on the Scanner and the reproducing spot on the display are produced by the master clock and sync generator. These signals and others are distributed throughout the EyeCom over the timing bus. All timing signals are derived from a master clock by means of digital counters that divide each line into 640 parts and count the lines through each frame. The instantaneous digital values in the counters are placed on the timing bus to provide the X and Y positions of the scanning spot. These positions or addresses represent 307,200 points or pixels in the picture: X-values range from 0 to 639 and Y-values from 0 to 479 (Y-values up to 511 are utilized in the output display refresh memory).

#### 5. SCANNER AMPLIFIER CHAIN

Prior to digitization, the video signal is processed by



the video amplifier chain which performs three functions for signal conditioning. First, it corrects the signal to compensate for "shading", i.e., changes in sensitivity of the camera tube across the image plane. Second, it amplifies the signal, linearly or logarithmically, to provide the desired relation between image brightness and the digitized Z-value. Third, it adjusts the sensitivity and offset of the video signal so that the digitized Z-values cover the desired image brightness range. The display control contains the setup, the zero, and the range registers which together control the signal conditioning circuits. The video signal from the video amplifier chain reproduces the conditioned Scanner signal on the display screen when it is selected by the video mixer and switch.

## 6. DIGITIZERS

The output of the video amplifier chain is digitized by the High-Resolution and Real-Time Digitizers. Data from the High-Resolution Digitizer is placed in the Z-register. The data bus provides computer access to the Z-register through the interface controller. The data from the Real-Time Digitizer occurs at too high a rate for transfer directly to the computer; consequently, a special digitizer data bus transfers this data to the refresh memory.



## 7. GRAPHICS AND PICTURE DISPLAY

The video signal that produces the Graphics and Picture Displays is generated from the refresh memory. This solid-state random-access memory has the capability to store a complete digitized field. The refresh memory is continuously read in synchronism with the sync generator, pixel by pixel, 320 times along each line for every one of the 240 lines, except when the data is being written. As the pixels are read, they are converted to graphics video signal which reproduces the stored picture on the display screen when selected by the video mixer and switch.

The data in the refresh memory can be displayed in two ways as selected by the interface controller. In the Picture Display mode, the 5 bits of each pixel are converted to a 32 level video signal that produces a picture on the display screen with 32 levels of gray. In the Graphics Display mode, 4 bits of each pixel are converted to produce a 16 gray-level picture while the other bit overrides the picture data to produce white graphics. Thus, line drawings and graphs can be superimposed on the display without destroying the picture data.

The Graphics and Picture Displays are produced by the video signal from the refresh memory. The memory is constructed from refresh memory planes, Figure 2, which are designed to be continuously read in synchronism with the television scan. The digital data from the memory is converted to a video signal that modulates the intensity of the



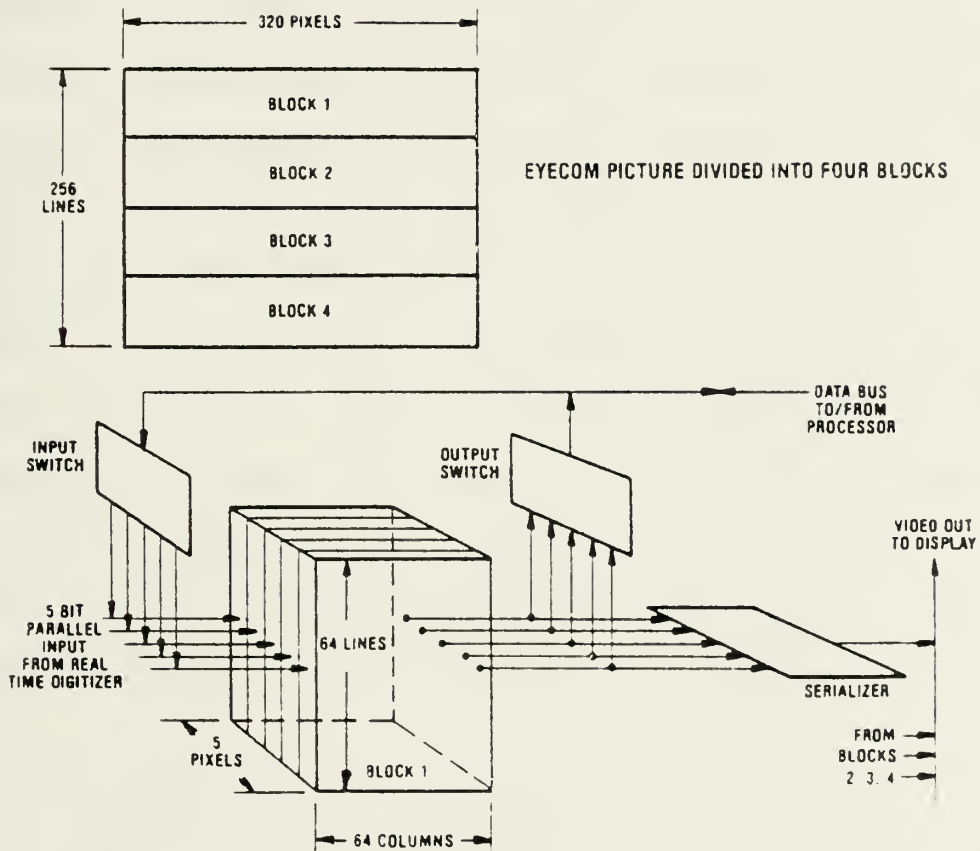


reproducing spot (see System Diagram, Figure 1) and reproduces the stored picture on the display.

Each memory plane has the capacity to refresh a 1-bit gray scale with a resolution of 320 pixels per horizontal scan line and up to 256 lines (240 lines are displayed, the other 16 may be displayed by scrolling the picture). Since a 1-bit gray scale is either black or white, a single memory plane is used to produce the Graphics Display. Four more memory planes are used in the EyeCom to produce the Picture Display which has a four-bit gray scale or 16 gray-levels. The graphics memory plane may be switched under computer control so that it forms a fifth, least-significant bit of the Picture Display. In this latter mode of operation, there is no Graphics Display and the Picture Display has twice the number of gray levels: 5 bits or 32 gray-levels.



Figure 2. MEMORY PLANE ORGANIZATION





## 8. MEMORY PLANE ORGANIZATION

Each memory plane contains four identical blocks of solid-state random access memory as shown in Figure 2. The storage capacity of each block is 640x320 bits contained in five integrated-circuit packages, 4096 bits per package.

The blocks are organized into 64 columns and 64 lines, Figure 2. The 64 lines correspond to the horizontal lines in the picture. Each of the 64 columns contain 5-bit words which are either read or written in parallel. A serializer provides a high-speed 1-bit output to refresh the display. An input and output switch provides 1-bit communication to the data bus for input-output to the digital processor. The switches and serializer are repeated for each of the four blocks in the memory plane. They are connected to provide a single bus for input and output to the digital processor and a single high-speed video output. Similarly, the 5-bit parallel input is connected to all four blocks. The input accepts the digitized picture data from the Real-Time Digitizer.

As the organization indicates, a bit or pixel in the memory plane is addressed by block, line, column, and one of the bits. The structure of the address is as follows:

- a) TV line address (3 bits): 2 bits to designate the block and 6 bits within the block; and,
- b) Pixel address across a TV line (9 bits): 6 bits to designate the column and 3 bits to



select one of 5 bits in the memory board.

Ten control lines are provided to control writing, reading, timing, and refreshing the memory chips. The bits are continuously read to produce the serial output data to refresh the display. Reading is momentarily stopped for 1.8 microseconds when the processor writes a single bit. When a digitized picture is written into memory from the Real-Time Digitizer, reading stops while the memory is loaded during the scanning rate of 1/60 of a second at a data rate of 12.8 megabits per second.

#### 9. ALPHANUMERIC DISPLAY

The video signal that produces the Alphanumeric Display is generated by the alphanumeric memory. This memory stores 24 lines of 80 ASCII characters each. Recirculating in synchronism with the sync generator, the ASCII codes are sequentially converted to alphanumeric video signals by the character generator.

#### 10. INTERFACE CONTROLLER

All EyeCom functions are controlled through the control





bus. This bidirectional bus transmits control signals to the displays, digitizers, and registers from the interface controller. It also sends status signals from the EyeCom internal units to the interface controller. Data is transmitted throughout the EyeCom on the data bus which transfers picture data to and from the alphanumeric memory, the refresh memory, and the High-Resolution Digitizer. The pixel location or address for the High-Resolution Digitizer and refresh memory is also transmitted through the data bus to the X and Y registers.

The interface controller connects the computer data bus, e.g., the Unibus of the PDP-11/50, to the control and data buses so that the computer program has complete control of the EyeCom and access to the picture data. The LA-30 Dec-writer Keyboard connects independently to the interface controller.

## 11. JOYSTICK

The Joystick and Joystick Control generate video for the Joystick Cursor which is displayed on the display screen under control of the video mixer and switch. The sync generator counters contain the instantaneous X-Y address of the reproducing spot on the display.







## 12. CONTROL AND DATA REGISTERS

The EyeCom is programmed through the fifteen registers shown in the System Diagram, Figure 1, and listed in Figure 3. The standard interface controller provides control of the EyeCom from Digital Equipment Corporation PDP-11/50.

The PDP-11/50 controller connects directly to the Unibus (input-output bus) and operates as a standard I/O device using the addresses shown in Figure 3 for each of the EyeCom registers. The EyeCom digitizers and displays are controlled through the registers as described in the following sections:

### 12.1. Command/Status Register

This register initiates the execution of all functions of the EyeCom. The command is executed as soon as the command register is loaded from the computer by a "MOVE" instruction. Commands that require significant time for execution provide an interrupt on completion if the appropriate interrupt enable bit is set in the command register or, in the case of the Keyboard and Alphanumeric Display, their status registers. With certain exceptions, a single bit in the command register is set corresponding to the function that is to be initiated. No more than one function can be initiated at one time. Certain auxiliary bits are set along with the command bits to provide modified commands such as increment and interrupt enable. The command/status register may be read to indicate the status of the EyeCom.



## 12.2. Display Register

Any one or more of the five EyeCom displays can be shown under the control of the display register. This register also controls whether the video from the Scanner is to be processed linearly or logarithmically. Any combination of bits may be set to superimpose the displays.

## 12.3. X, Y, and Z Registers

The address of the pixel to be digitized by the High-Resolution Digitizer or loaded into the refresh memory is contained in the X and Y registers. The Z-register contains the digitized data from the High-Resolution Digitizer. The Z-register is loaded with data from the computer when outputting data to the Picture Display.

## 12.4. Setup, Zero, and Range Registers

These three registers control the video amplifier chain for the proper black reference level and the desired zero and range of the digitization. The proper use of these registers is described in the Calibration Section (13).

## 12.5. Keyboard Status and Data Registers

The Keyboard data register is loaded when a key is depressed. Each key generates the standard 7-bit ASCII code for each character, upper and lower case.

## 12.6. Alphanumeric Registers

Four registers control the Alphanumeric Display. The next ASCII character is automatically written when it is loaded into the data register. The 6-bit ASCII code specifies capital letters, numerals, and special characters. The





position of the characters to be written is provided by the X-Y address which is contained in two bytes of the same word. The X-Y address is incremented after each character is written. Each character is written into the alphanumeric refresh memory and displayed immediately after the data register is loaded.

#### 12.7. Graphics Data Register

The data for the Picture and Graphics Displays is loaded in this register. Five refresh memory planes are provided within the EyeCom. The four most significant bits drive the Picture Display. The least significant bit drives the Graphics Display.

### 13. CALIBRATION

The relationship between the digitized Z-values and the brightness of the image is adjustable and can be calibrated using standard light sources or optical density tablets. The adjustments are made under program control using the setup, zero, and range registers. These registers control the video amplifier chain to produce the proper video signal for the digitizers as follows:

SETUP. The digital value, 0 to 255, in this register sets the proper black level of the Scanner signal to produce linear operation of the amplifier chain in the linear mode



and correct logarithmic operation in the log mode.

ZERO. The value in this register, 0 to 255, determines the video level corresponding to a digitized Z-value of zero.

RANGE. The value in this register, 0 to 63, determines the range of video levels covering the span from a Z-value of zero to a Z-value of 255.

The setup register is adjusted by first providing a black picture from the Scanner by capping the lens. A computer program is then used to repeatedly digitize a representative sample of pixels on the image in the linear mode and with the normal values in the zero and range registers, 0 to 63, respectively. After each digitization, the setup register is loaded with a new number until the averaged digitized Z-values lie near zero. The black level is then properly established. The black level is determined in part by the dark video signal from the Scanner camera tube and can change with temperature of the tube. The setup program is usually run prior to each digitization to insure the correct setting.

Having established the black level by means of the setup register and the brightest white level by the Scanner f-stop, the zero and range registers may now be changed to digitize only that portion of the image gray scale of interest to the user. The zero Z-value may be raised to any point on



the video signal by loading the zero register with the appropriate value. The zero Z-value position referred to as normal Z-value scale is given by the zero register value.

When the zero register value is raised, the range register value is usually reduced to bring the 255 Z-value level within the range of normal Z-values. When the range register is at the minimum value, zero, the range of Z-values are compressed and correspond to a range of 51 on the normal Z-value scale. The new zero,  $Z_0$ , and full scale,  $Z(255)$ , values expressed on the normal Z-value scale is given by the formulas:

$$Z_0 = \text{value in Zero Register}$$

$$Z(255) = Z_0 + 51 + (204/63) * R$$

where R is the value of the range register, 0 to 63, stored in the high-order 6 bits of the 8-bit register.

The setup programs use the Joystick Cursor to point out the brightest and darkest image areas of interest. The zero and range registers are set with correct values to place the darkest area at a Z-value of zero and the brightest area at a Z-value of 255. When this program is used with standard bright and dark references, then the relationship between Z-value and brightness is established providing calibration of the digitized values.



## 14. DIGITIZING THE PICTURE

As described earlier, the sync generator contains digital counting circuits for dividing each scan line into 640 parts and counts 480 lines in each frame. The instantaneous digital values representing the X and Y position of the scanning spot are available to the digitizer control on the timing bus. The instantaneous voltage from the video amplifier chain represents the brightness of the image at each X-Y position. This brightness value is sampled and digitized by the High-Resolution Digitizer under computer control.

The X-register and Y-register are loaded from the data bus with the location of the pixel to be sampled and digitized. The X and Y registers are continuously compared with the spot position as provided by the sync generator counters on the timing bus. When the scanning spot reaches the stored X-Y address, that is, the desired position, the address comparator issues a sample pulse to the High-Resolution Digitizer. Commands to sample and digitize a pixel are received by the digitizer control from the interface controller through the control bus. The sampled video is digitized into an 8-bit Z-value and stored in the Z-register where it can be accessed by the computer through the controller.

### 14.1. Normal Mode

The television picture is digitized as follows. The digital computer under program control first loads the





X-register and then the Y-register with the picture location to be digitized. The computer then issues an output instruction to transfer a digitize command to the digitizer control which activates the busy signal and tells the controller that the digitizing sequence is in process. When the scanning spot reaches the X-Y position, a sample pulse is issued and the pixel is digitized. The Z-value is placed in the Z-register. The busy signal is then reset and the computer interrupt is set. The computer program now transfers the Z-value to computer storage, completing digitization of the pixel. Subsequent Z-values can be immediately accessed and digitized by repeating the above sequence with different X-Y addresses.

#### 14.2. Sample and Increment Mode

The computer places the EyeCom in this mode by a modified digitize command which also initiates the digitizing sequence. The sequence is identical to the normal mode described previously, except that the X-Y address registers are incremented after every Z-value is obtained. The Y-register is incremented first causing the next address to fall two pixel locations vertically below the previously digitized pixel. Because the scanning lines are interlaced, this location is reached by the scanning spot during the next horizontal line period.

When digitize and increment commands are issued repeatedly, all even-numbered pixels in a vertical line are first digitized followed by all odd-numbered pixels. After a com-



plete vertical line is sampled, the X-register is automatically incremented and another vertical line of pixels digitized. In this way all the pixels in the image are digitized at 15,750 pixels per second.

#### 14.3. Digitizer Cursor

When the spot position from the sync generator compares with the X-Y address, the digitizer control generates a video signal that intensifies the television display spot. This produces a bright cursor showing the X-Y location on the picture. Every time the computer changes the X-Y address, this Digitizer Cursor appears to move across the picture to a new location. If digitize commands are given with each X-Y address, the Digitizer Cursor shows the points being digitized and serves as a monitor of the program operation. If the computer only outputs the X-Y addresses, the cursor will move under computer control to "draw" reference lines and points. The program can provide simple graphical information to the user in this manner.



## APPENDIX C

### RAMTEK PSEUDOCOLOR PROGRAM

```
/* Program to display a RSX-11M Picture File on the Ramtek  
Color Raster Scan Display using Pseudocolor by assigning a  
unique color to each of the possible sixteen gray-levels of  
a pixel. */
```

```
main() {
```

```
    int a[16],blk,j,k,n,of,mask,*pic;
```

```
    char in[4][512],out[240][2],p1,p2,p3,p4,*file;
```

```
    float x;
```

```
    ramtek();
```

```
    erase();
```

```
/* RAMTEK COLOR LOOKUP TABLE */
```

```
a[0]=triple(00,00,00);           //BLACK  
a[1]=triple(15,00,08);          //VIOLET  
a[2]=triple(04,00,12);          //PURPLE  
a[3]=triple(15,00,15);          //MAGNETA  
a[4]=triple(12,04,15);          //PUCE  
a[5]=triple(08,08,15);          //PINK  
a[6]=triple(00,00,15);          //RED  
a[7]=triple(00,12,15);          //ORANGE
```



```
a[8]=triple(00,15,15);           //YELLOW
a[9]=triple(00,15,08);           //YELLOW-GREEN
a[10]=triple(00,15,00);          //GREEN
a[11]=triple(15,08,00);          //GREEN-BLUE
a[12]=triple(08,15,00);          //BLUE-GREEN
a[13]=triple(15,15,00);          //CYAN
a[14]=triple(15,00,00);          //BLUE
a[15]=triple(15,15,15);          //WHITE

/* Load COLOR LOOKUP TABLE */

clrtbl(10,a);

/* RAMTEK COLOR TABLE */

colort(10);

/* Access EyeCom Picture File */

file = "pic.rsx";

/* Open UNIX Picture File */

pf = open(file,0);

/* Set Complex Data Mode */

setmode(3,0);
mask = 0017;
x = 10.0;
```





```

/*      Pack Picture File      */

for(blk=0;blk<640;blk=blk+4)
{
    read(pf,&in,2048);
    for(j=0;j<480;j=j+2)
    {
        n = j/2;
        p1 = (in[0][j]&mask)<<4;
        p2 = in[1][j]&mask;
        out[n][0] = p1|p2;
        p3 = (in[2][j]&mask)<<4;
        p4 = in[3][j]&mask;
        out[n][1] = p3|p4;
    }
}

/*      Output Packed Picture File To RAMTEK      */

strtxy(x,0.0);
for(k=0;k<240;k++)
{
    data(&out[k],2);
}
x = x + 0.5;
}
}

```



## LIST OF REFERENCES

1. Bernstein, Ralph, Digital Image Processing for Remote Sensing, The Institute of Electrical and Electronics Engineers Inc., 1978.
2. CSP Inc., Introduction to the MAP Series Models 100, 200, and 300, 1975.
3. CSP Inc., Marco Arithmetic Processor Systems (MAP-100/200/300) Programmer's Reference Manual, May 1977.
4. Digital Equipment Corporation, RSX-11M Beginner's Guide, 1977.
5. Digital Equipment Corporation, PDP 04/34/45/55/60 Processor Handbook, 1978.
6. Drexel University Center for Image Processing and Pattern Recognition, Modification of RT-11 Image Processing Programs for RSX-11M, by P.A. Wronski and P. Lafollette, 1978.
7. Nessler, Roger L., The Design of a User Interactive for a Color, Raster Scan Graphics Device, Master Thesis, Naval Postgraduate School, 1976.
8. Spatial Data Systems, Inc., Computer Eye Handbook of Image Processing, 1975.
9. Spatial Data Systems, Inc., EyeCom Handbook, 1977.
10. Spatial Data Systems, Inc., Image Processing Programs for PDP-11, 1975.
11. Visual Educom Inc., Instruction Manual for Television Camera Type 606 and 606A Part (Model) Number 100606, 6 May 1970.



INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center ATTN: DDC-TC Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Department Chairman, Code 52 Department of Computer Science Naval Postgraduate School Monterey, California 93940	2
4. Computer Laboratory, Code 52ec Department of Computer Science Monterey, California 93940	2
5. Professor George A. Rahe, Code 52ra Department of Computer Science Naval Postgraduate School Monterey, California 93940	1
6. Asst Professor Lyle A. Cox, Code 52cl Department of Computer Science Naval Postgraduate School Monterey, California 93940	1
7. Professor Robert Renard, Code 63rd Department of Meteorology Naval Postgraduate School Monterey, California 93940	1
8. Captain Donald Cain Program Manager AFIT/CIDD Wright-Patterson AFB, Ohio 45433	1
9. Captain Thomas M. Costello Department of Earth, Space and Graphic Science United States Military Academy West Point, New York 10996	1
10. Captain Robert C. Vogler 3708 Greene Ave. Omaha, Nebraska 68147	1









Thesis  
C75616 Costello  
c.1

183369

Implementation of a  
picture digitizer and  
display for image pro-  
cessing operations.

Thesis  
C75616 Costello  
c.1

183369

Implementation of a  
picture digitizer and  
display for image pro-  
cessing operations.



3 2768 002 08968 2

DUDLEY KNOX LIBRARY