

Reference

NBS
PUBLICATIONS

NAT'L INST. OF STAND & TECH
A11106 034049

NBSIR 85-3207

User's Manual for Division 746's Image Processing System

Dale P. Bentz
Jonathan W. Martin
Mary E. McKnight
Edward J. Embree

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Center for Building Technology
Gaithersburg, MD 20899

and

Martin E. Batts

Paratech
3201 Porter St., NW
Washington, DC 20008

July 1985



U.S. DEPARTMENT OF COMMERCE

NATIONAL BUREAU OF STANDARDS

QC
100
U56
85-3207
1985

NBSIR 85-3207

USER'S MANUAL FOR DIVISION 746'S IMAGE PROCESSING SYSTEM

Dale P. Bentz
Jonathan W. Martin
Mary E. McKnight
Edward J. Embree

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Center for Building Technology
Gaithersburg, MD 20899

and

Martin E. Batts

Paratech
3201 Porter St., NW
Washington, DC 20008

July 1985

U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, *Secretary*
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, *Director*

ABSTRACT

An image analysis system has been developed which allows the user to evaluate images in either an interactive or a batch mode. This manual provides instructions for accessing the imaging system (hardware and software) and describes the structure and function of each of the available commands. The imaging software is for an 80826 based minicomputer operating under a multi-user operating system with five imaging boards. The interactive run time environment is menu driven. To execute the imaging system in batch format, commands of a specific structure are placed into a datafile from which they are subsequently read and executed one at a time.

Keywords: Batch format; convolution; frame buffer; graylevel;
image analysis; image enhancement; imaging program;
pixel

TABLE OF CONTENTS

	Page
Introduction to the Imaging System.....	1
Introduction to the Imaging World.....	4
What is an Image.....	6
Commands and Structure.....	8
Appendices	
Definitions.....	26
Examples of Batch Files with Explanation.....	27

Introduction to the Imaging System

The imaging system includes a minicomputer, imaging boards, a stepper motor controller, a color intensity controller, a printer, and two cameras. The computer is an 80286 based Intel^{*} minicomputer with one megabyte of user memory, a Winchester 35Mb hard disk, an 8" floppy disk drive, and an external 9 track tape drive. The computer can support several users simultaneously.

The minicomputer contains five imaging boards: three frame buffer boards (FB-512), one analog processor board (AP-512), and one arithmetic logic unit board (ALU-512). The AP-512 board is used to control color on the monitor screen via lookup tables, to interact with the cameras and monitor, and to digitize the camera signal to eight bits. The FB-512 boards temporarily store images where they can be easily manipulated. The FB-512 system consists of a master 8-bit board (FB1) and two other boards which are linked as a 16-bit pair (FB0). Each FB-512 board can hold one image. The ALU-512 board provides the user with high speed image processing. It is used for convolutions and for making rapid transfers between frame buffers (i.e. moving frozen images from one FB-512 board to another).

*INTEL 80286/380 minicomputer. Identification is made solely to define the system and not to endorse a particular computer.

The system includes many peripherals, some of which have been specifically developed or modified for this imaging system. The basic peripherals include a printer, a color RGB monitor to view images, and an NTSC video camera. Specialized peripherals include the thermography camera, a stepper motor controller, which locates the camera in the x-y plane, and a color intensity controller used in mapping specific colors to certain grayscale ranges. Only one camera can be used at a time. This camera is attached by connecting the co-axial cable between the AP-512 board and the desired camera input.

The stepper motor controller can be used either manually or may be computer controlled from within the imaging program. The x-y position of the camera is controlled by sending a series of pulses to two stepper motors (one each for the x and y directions). Each pulse moves the camera a known amount in the desired direction. Limit switches prevent the camera from moving beyond the physical limits of the apparatus. Manually, the user can control the rate at which motion is achieved by dialing in the desired rate. In the computer controlled mode, the user simply inputs the coordinates of the desired location and the computer handles the rest.

The color intensity controller is used to assign color to various grayscale ranges. Once this option is selected from the imaging menu, the user interactively converts a grayscale range to a corresponding red, green, and blue color combination by

adjusting the color-coded dials on the front of the color intensity controller. The signals from the color intensity controller are sent to the computer via an A/D board. Once received by the computer, the signals are transformed into a color lookup table (see page 6) which is used in displaying the color image on the monitor.

Introduction to the Imaging World

While the previous section concentrated on the fundamental hardware devices, this section will present the basic procedure for executing the imaging programs as well as a brief description of a few of the operating system commands.

The first step in performing image analysis is to power up the system. This is accomplished by applying power first to the peripheral chassis (i.e. the box with the disk drives) and then to the system processor box. When turned on, the computer will respond with an asterisk (*) to which the user should respond by typing a capital U within twelve seconds to initiate the system self-tests. If the processor board is functioning properly, the user will next be instructed to enter either a period or a carriage return and should respond with a return. This will cause the computer to load and check the installed memory and disk drives. After completing the self-tests, the computer will initiate the bootstrap loader, which automatically loads the operating system and configures the computer hardware. At this time, the user may turn on and configure any desired peripherals such as a printer or tapedrive.

After the system tests are completed, the computer will boot up and execute an initial program. At this time the user is at the operating system level and has several options. One option is to perform file manipulation, program compilation, and other operating system commands within the current environment. The other option is to enter the image analysis system by

typing

```
:P:IMAGE.
```

This commands loads and executes the imaging task.

If the user wishes to execute IMAGE using a batch file containing commands to be implemented, they should type

```
:P:IMAGE filename
```

instead of the above.

Some of the more important operating systems commands are the DIR, DELETE, LPRINT, and COPY commands. DIR is short for directory and is used to print a listing of all files in the current home directory or in some user specified directory (e.g. DIR :SD: lists all files in the system directory). DELETE is used to delete files (DELETE filename). LPRINT is used to print the specified file to the printer with line numbers (LPRINT filename). Finally, COPY is used to copy one file to another file (COPY filename1 TO filename2). For further information on these and other operating system commands, one should refer to the INTEL Operator's Reference Manual.

The imaging programs are currently located in the directory :P: (an alias for :sd:user/world/prog/fortran). Images are stored in a further subdirectory, :P:IMAGES. Convolution filters are stored in a subdirectory called :P:FILTER. Lookup table (LUT) files are stored in the directory :P: under filenames of the form *****.LUT .

What Is An Image

To properly use the imaging system, one must have a basic understanding of what comprises an image. The picture presented on the monitor is a digitized image of the camera output. The screen is divided into discrete units (pixels), with 512 pixel resolution in the x-direction and 256 in the y-direction. Each pixel is assigned a grayscale value between 0 (black) and 255 (white) based on the incoming camera signal. This graylevel assignment is hardware controlled by a potentiometer on the AP-512 board and should be adjusted only when necessary.

Each pixel can be sampled by moving a pointer to any desired pixel location and reading its grayscale value (as can be done using the command CROSSHAIR [page 22]). This procedure is used in many of the imaging routines. In addition, one can assign a new grayscale value to any pixel in a similar fashion. This method is used in the procedures GREYSCALE and CROSSHAIR.

To produce images in color, one simply assigns to each grayscale value a color triple (red, green, blue) with each of the three values between 0 and 255 thus creating a lookup table (LUT). The incoming image grayscale values for each pixel are passed through the AP-512 board LUT where they are assigned a RGB combination so that they appear on the monitor in color.

The key to successful imaging is that the objects of interest (whether they be corrosion spots or cracks, etc.) have a grayscale value which consistently differs from that of the local background. Although some "enhancement" is possible, it is not possible to image objects for which no detectable graylevel difference is present.

Commands and Structure

The imaging system has many commands, some requiring a series of parameters. In the following pages, the commands will be grouped according to function and each one discussed in detail. The commands can be executed interactively and/or from a batch datafile. Interactively, program execution is menu driven with the user being prompted to make appropriate selections. In batch form, the commands are simply entered into a file in the proper format as listed in this section and will be executed one by one as prescribed. The following is an alphabetical listing of all commands available in the imaging system.

AREA	MAP_GREY_TO_RGB
BOX	MIX
CONVOLVE	MOVE
CROSSHAIR	POSITION_WRITE
DEFECT_AREA	PRINT
DIFFERENTIAL	REGISTER
DISPLAY	RETRIEVE
DISPLAY_TEMPORAL_AVERAGE	SAVE
EXIT	SAVELUT
FRAME_BUFFER	SET_CONTRAST_LEVELS
FREEZE	SMOOTH
GETLUT	STOP
GREYSCALE	THRESHOLD
INITIAL	VARIANCE
INPUT	WAIT
LUT	

1. Image Input/Output Commands

A. SAVE

Format- SAVE

Operation- saves the current image to a specified file for subsequent retrieval

Batch form- SAVE 'PATH/FILENAME'

Algorithm- The pixels are read one horizontal line at a time and written sequentially to the specified filename.

Assembler routines- GETLIN

Applications- This command allows the user to store images as they are obtained or to store "enhanced" images after processing.

Note- One must first freeze the current image (see 3A page 12) before saving, unless it is already stationary.

B. RETRIEVE

Format- RETRIEVE

Operation- retrieves a previously saved image from a file and displays it on the image monitor

Batch form- RETRIEVE 'PATH/FILENAME'

Algorithm- The pixels are retrieved one line at a time and written to the frame buffer for display.

Assembler routines- ZEROPAN, PUTLIN

Applications- RETRIEVE provides the user with the capability to recall previously saved images for comparison with new ones or for further processing.

C. FRAME BUFFER MANIPULATION

Format- FRAME_BUFFER

Operation- transfers an image between frame buffers
 numbers 0 and 1 for quick save/retrieval

Batch form- FRAME_BUFFER integer

integer-1	save (copy FB1 to FB0-1)
2	recall (copy FB0-1 to FB1)
3	save (copy FB1 to FB0-2)
4	recall (copy FB0-2 to FB1)

Algorithm- The procedure uses the ALU board to transfer
 the images between FB boards.

Assembler routines- LOAD0TO2, LOAD2TO0, GRABONE, GRAB_ZERO
 LOAD0TO3,LOAD3TO0

Applications- FRAME_BUFFER allows the user to temporarily
 store the current image so it can be preserved
 for future use if a grayscale histogram
 or thresholding is desired (See 8E and 8G
 pages 22 and 23).

Note- Using this command before doing a convolution (See
 8A page 20) will have no effect since all of the
 frame buffers are used by the convolution
 routine.

2. LUT (Lookup table) Commands

A. GETLUT

Format- GETLUT

Operation- retrieves a stored color lookup table (LUT)
from file

Batch form- GETLUT integer, 'PATH/FILENAME'
integer- # of lut unit to retrieve on

Assembler routines- ACTLUT, VIDOFF, SALLUT, VIDON

Applications- GETLUT allows the system user to use
standard lookup tables by recalling
them from storage.

B. LUT activation

Format- LUT

Operation- activates the user specified lut (#1-4)

Batch form- LUT integer
integer- # of lut to activate

Assembler routines- ACTLUT

Applications- LUT allows the user to toggle between lookup
tables so that an image can be viewed
in graylevel format or in color.

C. SAVE Lut

Format- SAVE Lut

Operation- saves an interactively defined LUT to a file
for future use

Batch form- SAVE Lut integer, 'PATH/FILENAME'
integer- # of lut unit to be saved

Assembler routines- none

Applications- SAVE Lut allows the user to create standard
lookup tables which can be used repeatedly.

3. Image format and manipulation

A. FREEZE

Format- FREEZE

Operation- freezes the current image on the screen by grabbing one picture and holding it

Batch form- FREEZE

Algorithm- The image is "frozen" by passing it through the ALU board and grabbing one frame on output

Assembler routines- FREEZ1

Applications- FREEZE's major application is in freezing an image so that it can be saved or processed.

B. DISPLAY

Format- DISPLAY

Operation- un-freezes the image, that is, allows continuous display of the incoming camera input

Batch form- DISPLAY

Assembler routines- ALUPAS, GRAB1

Applications- DISPLAY simply restores the system to continuously updating the image via the camera input and is also used to cancel temporal averaging (See 3D page 13).

C. SET CONTRAST LEVELS (2-256)

Format- SET_CONTRAST_LEVELS

Operation- changes the number of contrast levels displayed in an image

Batch form- SET_CONTRAST_LEVELS integer
integer- # of levels to set (2-256)

Assembler routines- SELLUT, SETLUT

Applications- SET_CONTRAST_LEVELS changes the number of apparent graylevels, so that any combination from simple black and white to 256 variations of gray may be displayed.

Note- This command is only useful when displaying or when the image is a currently frozen one as compared to a retrieved frozen image.

D. DISPLAY TIME AVERAGE

Format- DISPLAY_TEMPORAL_AVERAGE

Operation- averages a specified number of consecutive frames together to form a new frame

Batch form- DISPLAY_TEMPORAL_AVERAGE integer
integer- # of frames to average

Algorithm- The ALU board is used to time average the incoming image.

Assembler routines- VIDOFF, ALUDTA, VIDON, END_ACQUIRE1, SHIFT11

Applications- DISPLAY_TEMPORAL_AVERAGE is used to enhance the image by improving the signal to noise ratio.

Note- DISPLAY will cancel this averaging, but FREEZE will not.

4. Camera Control Commands

A. MOVE

Format- MOVE

Operation- moves the camera attached to the motion controller to a specific x,y location

Batch form- MOVE intx,inty
 intx- new x-position(0-5000)
 inty- new y-position(0-9000)

Algorithm- The parallel port on an iSBC communications board(the board used to provide multiple user access to the system) is used to communicate with the motion controller through a series of pulses.

Assembler routines- MOVE

Applications- MOVE will be instrumental in repetitive analyses where the camera needs to repeatedly sample a number of known locations.

Note- Movements in the x,y plane are made relative to a home position. It is necessary to initialize(move) the camera to the home position prior to moving it. This is accomplished by using the INITIAL command (See 5A page 16).

B. POSITION WRITE

Format- POSITION_WRITE

Operation- outputs to the user the current camera position (i.e. x,y coordinates)

Batch form- POSITION_WRITE

Algorithm- The position is read from the fortran language variables XCAM, YCAM

Assembler routines- None

Applications- POSITION_WRITE reminds the user of the current camera location in case some time has elapsed since the last move.

C. REGISTER

Format- REGISTER

Operation- centers the camera about an identifying mark such as a cross without changing the global camera coordinates (XCAM,YCAM) so that the user can start with the camera in a known location

Batch form- REGISTER

Algorithm- The mark is identified by the difference in its graylevel from that of the background and centered by moving the camera the necessary amount in the x and y directions.

Assembler routines- MOVE,RDPIXEL

Applications- REGISTER allows the user to start with the camera in a known location so that new images can be compared with earlier ones taken at the same location.

5. Batch Only Commands

A. INITIAL

Format- INITIAL

Operation- clears the LUT registers, initializes the camera to the home position, etc.

Assembler Routines- INIT, CLRLUT, INITSP

Applications- INITIAL is a startup procedure which allows the batch mode user to wipe the slate clean and start the imaging system over.

B. EXIT

Format- EXIT

Operation- exits the batch mode and enables the user to interact with the imaging system via the menus.

Assembler routines- None

Applications- EXIT provides the user with a good method for executing a series of general commands, followed by more specific commands which can be implemented interactively.

C. WAIT

Format- WAIT

Operation- instructs the computer to pause for a desired number of centiseconds.

Batch form- WAIT integer
 integer- # of centiseconds to wait

Algorithm- A call is performed to the system command RQSLEEP to "sleep" for the desired length of time.

Assembler routines- AWAIT

Applications- WAIT provides the user with a method for inserting pauses into a batch run (e.g. to provide adequate time for the user to read text printed on the screen).

D. PRINT

Format- PRINT

Operation- prints a string of text

Batch form- PRINT 'string'

Assembler routines- None

Applications- PRINT can be used to send text from a batch execution file to the videoscreen advising the operator of any necessary changes in the operating environment.

E. INPUT

Format- INPUT

Operation- allows the batch user to input an integer from the keyboard.

Batch form- INPUT I#
 #- number of variable to be input

Assembler routines- None

Applications- INPUT allows the user to input numbers into a batch execution which can be used as counters, flags, and identifiers.

6. Miscellaneous Commands

A. STOP

Format- STOP

Operation- terminates the imaging program and returns the user to the INTEL operating system (if program is being executed interactively) or to the batch datafile if it has been exited using the EXIT command

Batch form- STOP

Assembler routines- None

Applications- Obvious.

Note- The imaging boards are left in their previous state (i.e. if image is frozen, it will remain frozen, if displayed continuously, likewise.).

B. INTERACTIVE COLOR CONTROL

Format- MAP_GREY_TO_RGB

Operation- enables the user to interactively create a color lut using the color intensity controller. This is accomplished by dialing in a grayscale range and a RGB color combination and repeating until the desired effect is achieved.

Batch form- MAP_GREY_TO_RGB

Algorithm- The grayscale range and RGB inputs are read using an A/D conversion board and then scaled within the assembly language routines. The user has the option of keeping the changes or returning the image to its original state.

Assembler routines- SELLUT, SETLLUT, RDRGB

Applications- MAP_GREY_TO_RGB allows the user to experiment with an image by dialing in certain grayscale ranges and highlighting them with color.

C. BOX

Format- BOX

Operation- allows the user to specify a box (i.e. an area of the imaging screen) within which certain imaging commands will be executed. Commands affected by BOX include Threshold, Area, Grey-scale, Variance, and Defect_Area (see pages 21-24).

Batch Form- BOX int1,int2,int3,int4
 int1- minimum x-value for box
 int2- minimum y-value for box
 int3- maximum x-value for box
 int4- maximum y-value for box

Assembly Routines- BOXSET

Applications- BOX enables the user to implement commands which will affect only a portion of the image on the screen, thereby allowing a degree of selectivity in the imaging process.

7. Image Analysis Commands

A. CONVOLUTION

Format- CONVOLVE

Operation- performs a convolution filter on the image stored on the screen. The filter is input from a user specified file.

Batch form- CONVOLVE 'PATH/FILENAME',integer
filename- name of file for filter
integer- choice for operation after convolution (check for change of sign, transform, etc.)

Algorithm- The ALU is programmed from assembly to perform the convolution. After the convolution is performed, the user has the option of further refinement, via a change of sign detector, a 0-255 graylevel rescaling, and other techniques.

Assembler routines- KONVOLVE, GRSCALL

Applications- CONVOLVE can be used in detecting edges present in the current image.

Note- The image must be frozen or stationary (see 3A page 12) before a convolution is performed or the results will be unpredictable.

B. AREA sum

Format- AREA

Operation- determines the number and percentage of pixels in a user specified grayscale range.

Batch form- AREA intlow, inthigh
 intlow- low grayscale value
 inthigh- high grayscale value

Algorithm- The pixels are sampled one by one and the cumulative sum of pixels in the given range is updated and stored.

Assembly routines- AREASUM

Applications- AREA can determine the percentage of an image covered by some type of object if that object is contained exclusively in some grayscale range.

C. MIX (mix frame buffers 0 and 1 - channels 2 and 0)

Format- MIX

Operation- mixes the images in the frame buffers in a user specified fashion (ands, ors, adds, or subtracts)

Batch form- MIX integer
 integer- # of selected operation
 1) A+B 2) A-B
 3) A or B 4) A and B

Algorithm- Once again the ALU board is used to mix the two frames in the desired fashion.

Assembler routines- FREEZE, IMPMIX

Applications- MIX finds application in superimposing one image upon another and in computing differences between images of the same object taken under different conditions.

Note- For this command to be useful, frame buffer FB0 must contain one of the two starting images. This can be accomplished by using the FRAME_BUFFER (See 1C page 10) command to load a desired image into FB0.

D. CROSSHAIR

Format- CROSSHAIR

Operation- allows the user to interactively locate a crosshair on the screen and read the pixel grayscale value at the center point, or to produce a cross-sectional histogram of the horizontal line grayscale values. The crosshair can be positioned on the screen through interactive use of the arrow keys. Size of moves is user controlled via a menu selection.

Batch form- CROSSHAIR (not meant to be used
in batch mode)

Algorithm- The crosshair is produced by Xoring (exclusive oring) each pixel with itself, so that the original pixel grayscale value is preserved. (i.e. a second XOR operation will restore it to its original value.)

Assembler routines- CROSTART, CROSSECT, RDPIXEL

Applications- CROSSHAIR has several applications. It can be used to determine the grayscale value of a given pixel or to determine the grayscale profile of a line of pixels.

E. THRESHOLD

Format- THRESHOLD

Operation- thresholds an image for a grayscale range as follows:
MINPIX < [pixel] < MAXPIX [pixel]= 0
otherwise [pixel]= 255

Batch form- THRESHOLD intmin,intmax
intmin= MINPIX
intmax= MAXPIX

Assembler routines- THRES1

Applications- THRESHOLD finds use in object detection since objects of interest can be highlighted as black against a white background.

F. DEFECT AREA

Format- DEFECT_AREA

Operation- searches and locates defects and returns sizes, centroids, and radii of gyration.

Batch form- DEFECT_AREA 'PATH/FILENAME',intl,intlo,inthi,
output device

filename- file for convolution filter

intl- choice after convolution

intlo- min. value for threshold

inthi- max. value for threshold

output device- device to output
results (file or printer)

Algorithm- The image is first convoluted to highlight objects of interest, then thresholded to enable the program to find defects by searching for pixels with a grayscale value of zero.

Assembler routines- CHECKHIT, HIT, STATS

Applications- DEFECT_AREA enables the user to determine the size of objects such as corrosion spots.

G. GRAYSCALE histogram

Format- GREYSCALE

Operation- produces a grayscale histogram superimposed on the currently displayed image

Batch form- GREYSCALE

Assembler routines- GRSCALE, ZEROPAN, GRPLOT

Applications- GREYSCALE allows the user to determine distinct grayscale ranges of interest by noting peak and valleys in the histogram.

Note- If you don't want to lose the original image use FRAME_BUFFER to store it in FB0 temporarily and then recall it from there after viewing the histogram.

H. SMOOTH image

Format- SMOOTH

Operation- smooths image in x and y directions.

Batch form- SMOOTH

Assembler routines- SMOOTH

Applications- SMOOTH can be used to eliminate some of the local noise present in images.

I. DIFFERENTIATE image

Format- DIFFERENTIAL

Operation- differentiates the current image by finding the maximum gradient at each pixel.

Algorithm- For each pixel, the surrounding eight pixels are sampled. The current pixel grayscale value is replaced by the maximum difference between the current pixel and the surrounding neighbors.

Batch form- DIFFERENTIAL

Assembler routines- DELTA, LOAD0TO2, GRSCALE, THRES1

Applications- DIFFERENTIAL provides the user an alternate method to detect edges since in the differentiated image edges will be brighter than the surrounding background.

J. GRAYSCALE STATISTICS

Format- VARIANCE

Operation- computes average graylevel value and standard deviation in graylevel for image on screen.

Batch form- VARIANCE

Algorithm- The sums of the pixel graylevel values and the graylevel values squared are computed in Assembly. From these, the average and standard deviation can be computed.

Assembly routines- AREAL

Applications- Several possible applications exist. VARIANCE can be used to check for imaging homogeneity or to possibly provide a measure of the surface profile (i.e. larger std. dev. = rougher surfaces).

Definitions

ANDING TWO IMAGES- In anding two images, the grayscale values for the same pixels in the two images are anded in binary (e.g. 01101011 and 11100110 becomes 01100010 when the two are anded). When all of the pixels are anded, a superposition is achieved; i.e. edges can be superimposed on the original image). The basic rules for anding two numbers are

0 and 0 is 0
0 and 1 is 0
1 and 0 is 0
1 and 1 is 1.

CONVOLUTION (filter)- A convolution filter is an $n \times n$ matrix which is applied centrally at each pixel of the original image. Its function is to determine the edges of objects in the image.

FILENAME (form)- A filename is composed of an eight (max) letter identifier followed optionally by a dot and a three letter extension(e.g. XIMAGE.ASM).

FRAME BUFFER- A frame buffer is memory allocated for storing all of the pixel grayscale values for a particular image. The imaging system currently contains three such frame buffers (the FB-512 boards).

GRAYSCALE RANGE- A grayscale range is a range of grayscale values characterized by a minimum and a maximum value (e.g. 10-100 is the grayscale range for all pixels with grayscale values between ten and one hundred).

LOOKUP TABLE- A lookup table is an input/output system where the output values are dependent on the input entry. IRS tax tables are an example of this where the output (the tax you owe) is dependent on the input (your income and number of dependents).

PIXEL- A pixel is an individual display unit. It is the smallest display unit which can be controlled. All monitor screens are comprised of a matrix of pixels.

ORING TWO IMAGES- Oring two images is similar to anding them except that the pixels are ored instead of anded(e.g. 01101011 and 11100110 becomes 11101111 when ored.). The basic rules of oring two numbers are

0 or 0 is 0
0 or 1 is 1
1 or 0 is 1
1 or 1 is 1.

Examples of Batch Files with Explanation

1) This first batch file initializes the system, retrieves a frame from file, finds the defects present in the image, outputs the results to the printer, returns to displaying continuously, and exits the imaging system.

```
INITIAL
RETRIEVE 'IMAGES/COATTT.42'
DEFECT_AREA 'FILTER/GAUSS25.DTA',3,140,255,':LP:'
DISPLAY
STOP
```

2) This second batch file initializes the system, moves the camera to position 120,480, freezes the display, stores the displayed image, returns to displaying continuously and exits to the interactive mode for further analysis.

```
INITIAL
MOVE 120,480
FREEZE
SAVE 'IMAGES/TESTPATT.IMG'
DISPLAY
EXIT
```


U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)		1. PUBLICATION OR REPORT NO. NBSIR 85-3207	2. Performing Organ. Report No.	3. Publication Date July 1985
4. TITLE AND SUBTITLE User's Manual for Division 746's Image Processing System				
5. AUTHOR(S) Dale P. Bentz, Martin E. Batts, Jonathan W. Martin, Mary E. McKnight, Edward J. Embree				
6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions) NATIONAL BUREAU OF STANDARDS DEPARTMENT OF COMMERCE WASHINGTON, D.C. 20234 and ParaTech 3201 Porter St. N.W. Washington, DC 20008			7. Contract/Grant No.	8. Type of Report & Period Covered
9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP)				
10. SUPPLEMENTARY NOTES <input type="checkbox"/> Document describes a computer program; SF-185, FIPS Software Summary, is attached.				
11. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here) An image analysis system has been developed which allows the user to evaluate images in either an interactive or a batch mode. This manual provides instructions for accessing the imaging system (hardware and software) and describes the structure and function of each of the available commands. The imaging software is for an 80826 based minicomputer operating under a multi-user operating system with five imaging boards. The interactive run time environment is menu driven. To execute the imaging system in batch format, commands of a specific structure are placed into a datafile from which they are subsequently read and executed one at a time.				
12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons) batch format; convolution; frame buffer; graylevel; image analysis; image enhancement; imaging program; pixel				
13. AVAILABILITY <input checked="" type="checkbox"/> Unlimited <input type="checkbox"/> For Official Distribution. Do Not Release to NTIS <input type="checkbox"/> Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402. <input checked="" type="checkbox"/> Order From National Technical Information Service (NTIS), Springfield, VA. 22161			14. NO. OF PRINTED PAGES 28 15. Price \$8.50	

