NAME

Exit Program -- Close program.

SYNOPSIS

exit

DESCRIPTION

Exit the main function with exit code 1

EXAMPLES

exit

NAME

List Buffers -- Print a list of all buffers in memory

SYNOPSIS

list

DESCRIPTION

All images and transformations of those images are saved in buffers. These buffers are displayed when **list** is called. Buffers can be written to files of ___ type.

EXAMPLES

n/a

SEE ALSO

n/a

NAME

Show Image in Buffer -- Display the contents of <buffer1> on the screen.

SYNOPSIS

display *buffer1*

DESCRIPTION

This routine allows users in a UNIX GUI environment to easily view any image previously read into a buffer.

buffer : The buffer containing the image to be displayed on the screen.

EXAMPLES

display buffer1;

SEE ALSO

n/a

NAME

Input Image -- Read from the filesystem into a memory buffer.

SYNOPSIS

read *imagename* into *buffer*

DESCRIPTION

The image located at imagename (relative to the CWD) is loaded into a memory buffer named *buffer.*
The separator “into” can technically be anything except a space or newline character.

imagename: Path of image file to be read
buffer : Name of memory buffer

EXAMPLES

Read ../../test_img.png into 1
Read cats.jpeg into box

SEE ALSO

Output Image (5)

NAME

Output Image -- Write the image in a buffer to a file

SYNOPSIS

write *buffer-name* into *new-image-name*

DESCRIPTION

Output Image allows the user to write a buffer containing an image to a file on disk. The separator “into” can technically be anything except a space or newline character.

buffer-name : Name of the image buffer to be written
new-image-name : Name of the file saved to disk

EXAMPLES

write buff1 into foo.png

SEE ALSO

Input Image (4)

NAME

Addition -- Do pixel-wise addition of two images with the same resolution.

SYNOPSIS

buff_dest = *buff1* **+** *buff2*

DESCRIPTION

The addition operator allows 2 buffers (buff1 & buff2) to be added together. They are then saved in the destination buffer, buff_dest.

EXAMPLES

destination = buffer1 + buffer2

NAME

Subtraction-- Do pixel-wise subtraction of two images with the same resolution.

SYNOPSIS

buff_dest = *buff1* **-** *buff2*

DESCRIPTION

The subtraction operator allows buffer2 to be subtracted from buffer1. They are then saved in the destination buffer, buff_dest.

EXAMPLES

destination = buffer1 - buffer2

NAME

Multiplication -- Do pixel-wise multiplication of two images with the same resolution.

SYNOPSIS

*buff_dest = buff1 * buff2*

DESCRIPTION

The multiplication operator allows buffer2 to be multiplied with buffer1. They are then saved in the destination buffer, buff_dest.

EXAMPLES

destination = buffer1 * buffer2

NAME

Division -- Do pixel-wise division of two images with the same resolution.

SYNOPSIS

buff_dest = buff1 / buff2

DESCRIPTION

The division operator allows buffer1 to be divided by buffer2. They are then saved in the destination buffer, buff_dest.

EXAMPLES

destination = buffer1 / buffer2

NAME

Brighten -- Adjustable brightening routine.

SYNOPSIS

brighten *buff_source* **into** *buff_dest* **by** *brighten_value*

DESCRIPTION

The image in buff_source will be brightened with an intensity according to the *brighten_value.* The brightened image will be stored into buff_dest. brighten_value is a number between 0 and 255.

EXAMPLES

brighten buff1 into buff2 by 5

SEE ALSO

Darken (11)

NAME

Darken -- Adjustable darkening routine.

SYNOPSIS

darken *buff_source* **into** *buff_dest* **by** *dark_value*

DESCRIPTION

The image in buff_source will be darkened with an intensity according to the *dark_value.* The darkened image will be stored into buff_dest. dark_value is a number between 0 and 255.

EXAMPLES

darken buff1 into buff2 by 5

SEE ALSO

Brighten (10)

NAME

Edge Detection -- Multiple edge detection algorithms in a single command.

SYNOPSIS

--method --style buff1 **into** *buff_dest*

DESCRIPTION

--method:

- Horizontal
- Vertical
- Combined

--style:

- Kirsch
- Prewitt
- Sobel

Based on the user's choice in *method* and *style,* the image in buff1 will be run through the edge detection algorithm of choice, with the result being stored in buff_dest.

EXAMPLES

Kirsch Combined in_buff into out_buff

SEE ALSO

Histogram Equalisation (13)
Sharpening (20)
Define Convolution Template (17)
Template Convolution (18)
3x3 Convolution (19)

NAME

Histogram Equalization -- Stretch the image histogram to use the entire intensity range.

SYNOPSIS

histeq *buff1* **into** *buff2*

DESCRIPTION

Uses statistical methods to normalize the contrast in an image, normally to increase the image's contrast.

buff1: Input Buffer
buff2: Output Buffer

EXAMPLES

histeq inbuff into outbuff

SEE ALSO

A list of related commands or functions.

NAME

Flip -- Flip images vertically and horizontally.

SYNOPSIS

flip *direction buff_dest*.

DESCRIPTION

direction: horizontal | vertical
buff_dest: Output Buffer

EXAMPLES

Some examples of common usage.

SEE ALSO

Rotate (15)

NAME

Rotation -- Allows rotation of an input image by a determined number of degrees.

SYNOPSIS

rotate *buff1* **by** *degrees*

DESCRIPTION

buff1: Input Buffer

degrees: Variable number of degrees to rotate between -360 and 360.

EXAMPLES

rotate my_buffer by -90 (Rotates by 90 degrees counterclockwise)

SEE ALSO

Flip (14)

NAME

Blurring -- Gaussian Blur Implementation.

SYNOPSIS

Blurr *buff1 radius sigma*

DESCRIPTION

buff1: Input Buffer
radius: Blurring radius variable
Sigma: Blurring standard deviation variable

EXAMPLES

Blurr myBuffer 5 2

SEE ALSO

Sharpen (20)

NAME

Define Convolution Template -- Save a convolution kernel in memory.

SYNOPSIS

define_template *template-name template-values* **tp =** *template-tp*

DESCRIPTION

template-name: Name for the template buffer
template-values: Values of the kernel in bracket notation (see example)
template-tp: Target pixel for convolution

EXAMPLES

define_template myTemplate [5 2] [0 6] tp = (0,0)

SEE ALSO

Convolve Template (18)

NAME

Convolve Template -- Convolve an image with a user-defined kernel.

SYNOPSIS

convolve_template *buff1 template-name*

DESCRIPTION

buff1: Input buffer
template-name: Selects the user-defined template to use by name

EXAMPLES

Convolve_template inputbuff myTemplate

SEE ALSO

Define Convolution Template (17)
3x3 Convolution (19)

NAME

3x3 Convolution -- One-line convolution command with template input

SYNOPSIS

convolve3x3 *template buff1*

DESCRIPTION

template: 3x3 matrix represented by 3 sets of brackets, each representing a row
buff1: Input buffer

The convolve3x3 function assumes a target pixel of (0,0), and gives the user a fast method of convolving with an image.

EXAMPLES

convolve3x3 [0 0 1] [0 1 0] [1 0 0] myBuffer

SEE ALSO

Define Convolution Template (17)
Template Convolution (18)

NAME

Sharpen Image -- One-line sharpening command with intensity input

SYNOPSIS

sharpen *intensity buff1*

DESCRIPTION

intensity: low | high
buff1: Input buffer

EXAMPLES

sharpen low myPicture

SEE ALSO

Blur (16)