# Assignment 1

Fatemeh (Hoorad) Abootalebi

## ROI

ROI or region of interest is a rectangular that select by specifying pixel location (X,Y) of the left-top pixel of ROI and (SX,SY) of the right-down pixel of ROI.

User can define at least 1 ROI per each image. There is no limit for number of ROIs.

ROIs should not have overlaps. All ROIs check by checkROI function. If there is an overlap, the process stops.

## Add Gray:

This function add the intensity to each pixel of grayscale images.

Value added to each pixel in ROI. If the result is greater than 255, it sets 255.

Parameters:

|  |  |  |
|---|---|---|
| src: | Address of source image | |
| tgt: | Modified image is saved in this address | |
| roi_number: | Number of defined ROI | |
| Input Vector: | value | This value add to intensity of all pixels |
| | ROI | Target ROI of image |

## Example

```
SampleImages\Gray\baboon.pgm OutputImages\baboon_add.pgm
add 2 70 0 0 100 400 100 300 150 500 500
```

| Image Address | SampleImages\Gray\baboon.pgm OutputImages\baboon_add.pgm | Address of input image Address of modified image |
|---|---|---|
| Function | add | Function name |
| ROI | 2 | Number of ROI |
| First Input | 70 | Value of first ROI |
| | 0 | X of first ROI |
| | 0 | Y of first ROI |
| | 100 | SX of first ROI |
| | 400 | SY of first ROI |
| Second Input | 100 | Value of second ROI |
| | 300 | X of second ROI |
| | 150 | Y of second ROI |
| | 500 | SX of second ROI |
| | 500 | SY of second ROI |

## Output

Add gray with 2 ROIs (ROI1: left, ROI2: right)



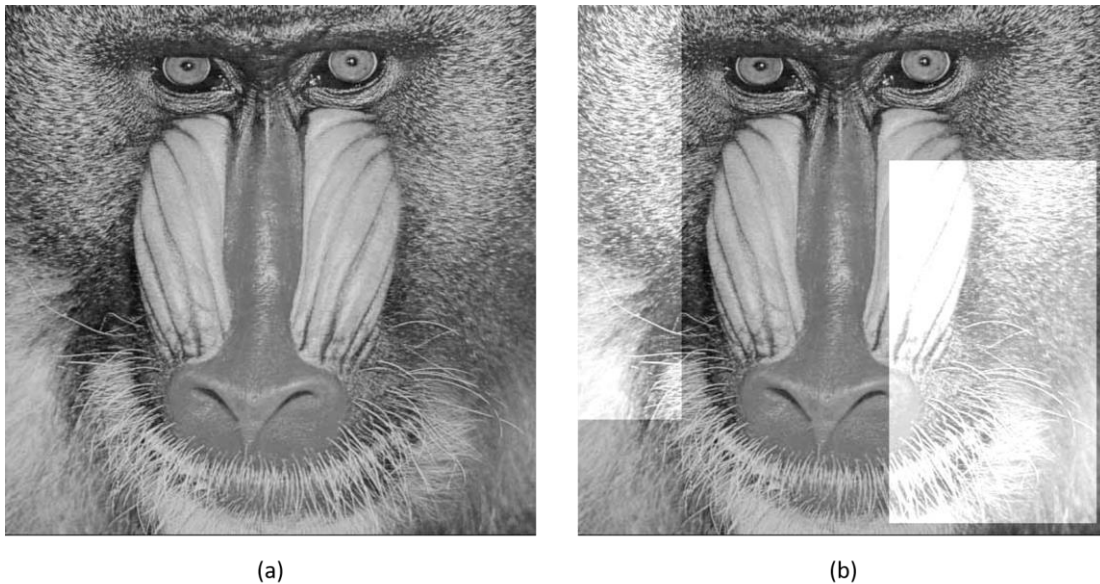(a)                                        (b)

*Figure 1 (a) Original Image (b) Modified Image*

(a) Original image

(b) Value = 70 for ROI1

    Value = 100 for ROI2

## Binarize:

This function creates a binary image from 2D grayscale image by replacing all values above the threshold with 255s and setting all other values to 0s.

### Parameters:

| | |
|---:|:---|
| src: | Address of source image |
| tgt: | Modified image is saved in this address |
| roi_number: | Number of defined ROI |
| Input Vector: | threshold    All pixels above the threshold set to 255, otherwise set to 0 |
| | ROI          Target ROI of image |

### Example

```
SampleImages\Gray\floor.pgm OutputImages\floor_binarize.pgm
binarize 3 120 0 0 500 100 160 100 200 300 300 89 420 150 500 360
```

| Image Address | SampleImages\Gray\floor.pgm | Address of input image |
| --- | --- | --- |
| | OutputImages\floor_binarize.pgm | Address of modified image |
| Function | binarize | Function name |
| ROI | 3 | Number of ROI |
| First Input | 120 | Value of first ROI |
| | 0 | X of first ROI |
| | 0 | Y of first ROI |
| | 500 | SX of first ROI |
| | 100 | SY of first ROI |
| Second Input | 160 | Value of second ROI |
| | 100 | X of second ROI |
| | 200 | Y of second ROI |
| | 300 | SX of second ROI |
| | 300 | SY of second ROI |
| Third Input | 89 | Value of third ROI |
| | 420 | X of third ROI |
| | 150 | Y of third ROI |
| | 500 | SX of third ROI |
| | 360 | SY of third ROI |

## Output
Binarize with 3 ROIs (ROI1: top, ROI2: middle, ROI3: bottom-right)



(a)　　　　　　　　　　　　　　　(b)

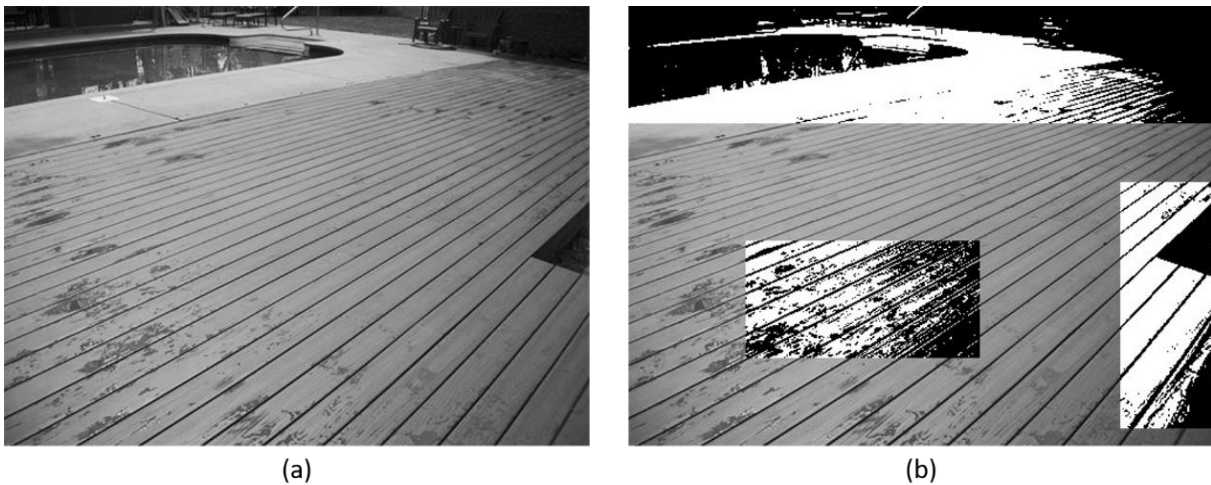*Figure 2 (a) Original Image (b) Modified Image*

(a)  Original image
(b)  Threshold = 120 for ROI1
     Threshold = 160 for ROI2
     Threshold = 89 for ROI2

## Brightness:

This function decrease or increase the intensity of each pixel based on the threshold.

If the intensity is larger than the threshold, then increase the brightness by value1. If the intensity is lower than the threshold, then decrease the brightness by value2. Otherwise there is no change.

## Parameters:

|  |  |  |
|---|---|---|
| src: | Address of source image | |
| tgt: | Modified image is saved in this address | |
| roi_number: | Number of defined ROI | |
| Input Vector: | threshold | All pixels above the threshold set to 255, otherwise set to 0 |
| | value1 | Increase the brightness by value1 of the pixel with the intensity larger than the threshold |
| | value2 | Decrease the brightness by value2 of the pixel with the intensity smaller than the threshold |
| | ROI | Target ROI of image |

## Example

```
SampleImages\Gray\lena.pgm OutputImages\lena_brightness.pgm
brightness 3 128 50 70 50 180 300 512 180 30 40 50 30 462 150 100
50 80 301 200 512 300
```

| Image Address | SampleImages\Gray\lena.pgm | Address of input image |
|---|---|---|
| | OutputImages\lena_brightness.pgm | Address of modified image |
| Function | brightness | Function name |
| ROI | 3 | Number of ROI |
| First Input | 128 | Threshold of first ROI |
| | 50 | Value1 of first ROI |
| | 70 | Value2 of first ROI |
| | 50 | X of first ROI |
| | 180 | Y of first ROI |
| | 300 | SX of first ROI |
| | 512 | SY of first ROI |
| Second Input | 180 | Threshold of second ROI |
| | 30 | Value1 of second ROI |
| | 40 | Value2 of second ROI |
| | 50 | X of second ROI |
| | 30 | Y of second ROI |
| | 462 | SX of second ROI |
| | 150 | SY of second ROI |
| Third Input | 100 | Threshold of third ROI |
| | 50 | Value1 of third ROI |
| | 80 | Value2 of third ROI |
| | 301 | X of third ROI |
| | 200 | Y of third ROI |
| | 512 | SX of third ROI |
| | 300 | SY of third ROI |

## Output
Modify brightness with 3 ROIs (ROI1: top, ROI2: left, ROI3: right)



(a)                                                      (b)

*Figure 3 (a) Original Image (b) Modified Image*

(a) Original image
(b) Threshold = 128 and Value1 = 50 and Value2 =70 for ROI1
    Threshold = 180 and Value1 = 30 and Value2 =40 for ROI2
    Threshold = 100 and Value1 = 50 and Value2 =80 for ROI2

## Smoothing:
Averaging is a special case of discrete convolution. For a $3 \times 3$ neighborhood, the convolution mask h is

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \text{ and for a } 5 \times 5 \text{ neighborhood is } \frac{1}{25}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Each kernel window convolve whit averaging matrix. The middle index of kernel matrix set to result of convolution.

The edge pixels remain the same as the original pixel value.

## Parameters:

|  |  |  |
|---|---|---|
| src: | Address of source image | |
| tgt: | Modified image is saved in this address | |
| roi_number: | Number of defined ROI | |
| Input Vector: | window | The window size, it could be 3 or 5 |
| | ROI | Target ROI of image |

## Example

```
SampleImages\Gray\tree.pgm OutputImages\tree_smoothing.pgm
smoothing 2 3 80 80 230 350 5 231 50 400 300
```

| Image Address | `SampleImages\Gray\tree.pgm` | Address of input image |
|---|---|---|
| | `OutputImages\tree_smoothing.pgm` | Address of modified image |
| Function | `smoothing` | Function name |
| ROI | 3 | Number of ROI |
| First Input | 3 | Window size of first ROI |
| | 80 | X of first ROI |
| | 80 | Y of first ROI |
| | 230 | SX of first ROI |
| | 350 | SY of first ROI |
| Second Input | 5 | Window size of second ROI |
| | 231 | X of second ROI |
| | 50 | Y of second ROI |
| | 400 | SX of second ROI |
| | 300 | SY of second ROI |

## Output

Smoothing with 2 ROIs (ROI1: left, ROI2: right)



(a)                    (b)

*Figure 4 (a) Original Image (b) Modified Image*

(a) Original image
(b) Window = 3 for ROI1
    Window = 5 for ROI2

## Color Brightness:

This function modify the intensity of each channel of each pixel based on the More-C.

The red channel multiplies by More-C ($R' = R_0 \times More - C$). The green and blue channels add by More-C ($G' = G_0 + More - C$ and $B' = B_0 + More - C$)

## Parameters:

|  |  |  |
|---:|:---|:---|
| src: | Address of source image | |
| tgt: | Modified image is saved in this address | |
| roi_number: | Number of defined ROI | |
| Input Vector: | More-C | Process each channel either add or multiple with More-C |
| | ROI | Target ROI of image |

## Example

```
SampleImages\Color\frog.ppm OutputImages\frog_brightness.ppm
brightness 3 100 100 0 230 200 10 250 30 480 260
```

| Image Address | SampleImages\Color\frog.ppm | Address of input image |
|---|---|---|
| | OutputImages\frog_brightness.ppm | Address of modified image |
| Function | cbrightness | Function name |
| ROI | 3 | Number of ROI |
| First Input | 100 | More-C of first ROI |
| | 100 | X of first ROI |
| | 0 | Y of first ROI |
| | 230 | SX of first ROI |
| | 200 | SY of first ROI |
| Second Input | 10 | More-C of second ROI |
| | 250 | X of second ROI |
| | 30 | Y of second ROI |
| | 480 | SX of second ROI |
| | 260 | SY of second ROI |

## Output
Modify brightness with 2 ROIs (ROI1: left, ROI2: right)



(a)                                                  (b)

*Figure 5 (a) Original Image (b) Modified Image*

(a)  Original image
(b)  More-C = 100 for ROI1
        More-C = 10 for ROI2

## Color Binarize:

This function modify each pixel to red, green, or white based on the threshold.

It calculate Euclidian distance each pixel in RGB space from (0, 0, 0) and named it as OTC.

If OTC is lower than TC, intensity of this pixel set to (255, 0, 0). If OTC is between TC and 2TC, intensity of this pixel set to (0, 255, 0). Otherwise it set to (255, 255, 255).

## Parameters:

|  |  |  |  |
|---|---|---|---|
| src: | Address of source image | | |
| tgt: | Modified image is saved in this address | | |
| roi_number: | Number of defined ROI | | |
| Input Vector: | TColor | | Euclidian distance threshold |
| | | CR | Defined channel red in RGB space |
| | C | CG | Defined channel green in RGB space |
| | | CB | Defined channel blue in RGB space |
| | ROI | | Target ROI of image |

## Example

```
SampleImages\Color\flower.ppm OutputImages\flower_binarize.ppm
cbinarize 4 150 10 10 10 0 0 100 100 200 0 255 0 230 80 430 280
100 20 180 20 0 270 100 374 200 0 255 0 430 310 500 374
```

| | | |
|---|---|---|
| Image Address | `SampleImages\Color\flower.ppm` | Address of input image |
| | `OutputImages\flower_binarize.ppm` | Address of modified image |
| Function | cbinarize | Function name |
| ROI | 4 | Number of ROI |
| First Input | 150 | TC of first ROI |
| | 10 | CR of first ROI |
| | 10 | CG of first ROI |
| | 10 | CB of first ROI |
| | 0 | X of first ROI |
| | 0 | Y of first ROI |
| | 100 | SX of first ROI |
| | 100 | SY of first ROI |
| Second Input | 200 | TC of second ROI |
| | 0 | CR of second ROI |
| | 255 | CG of second ROI |
| | 0 | CB of second ROI |
| | 230 | X of second ROI |
| | 80 | Y of second ROI |
| | 430 | SX of second ROI |
| | 280 | SY of second ROI |
| Third Input | 100 | TC of third ROI |
| | 20 | CR of third ROI |
| | 180 | CG of third ROI |
| | 20 | CB of third ROI |
| | 0 | X of third ROI |
| | 270 | Y of third ROI |
| | 100 | SX of third ROI |
| | 374 | SY of third ROI |
| Fourth Input | 200 | TC of fourth ROI |
| | 0 | CR of fourth ROI |
| | 255 | CG of fourth ROI |
| | 0 | CB of fourth ROI |
| | 430 | X of fourth ROI |
| | 310 | Y of fourth ROI |
| | 500 | SX of fourth ROI |
| | 374 | SY of fourth ROI |

# Output

Color binarization with 4 ROIs (ROI1: top-left, ROI2: middle, ROI3: bottom-left, ROI4: bottom-right)



(a)                                                     (b)

*Figure 6 (a) Original Image (b) Modified Image*

(a) Original image
(b) TC = 150 and C (10, 10, 10) for ROI1 and ROI4
    TC = 200 and C (0, 255, 0) for ROI2
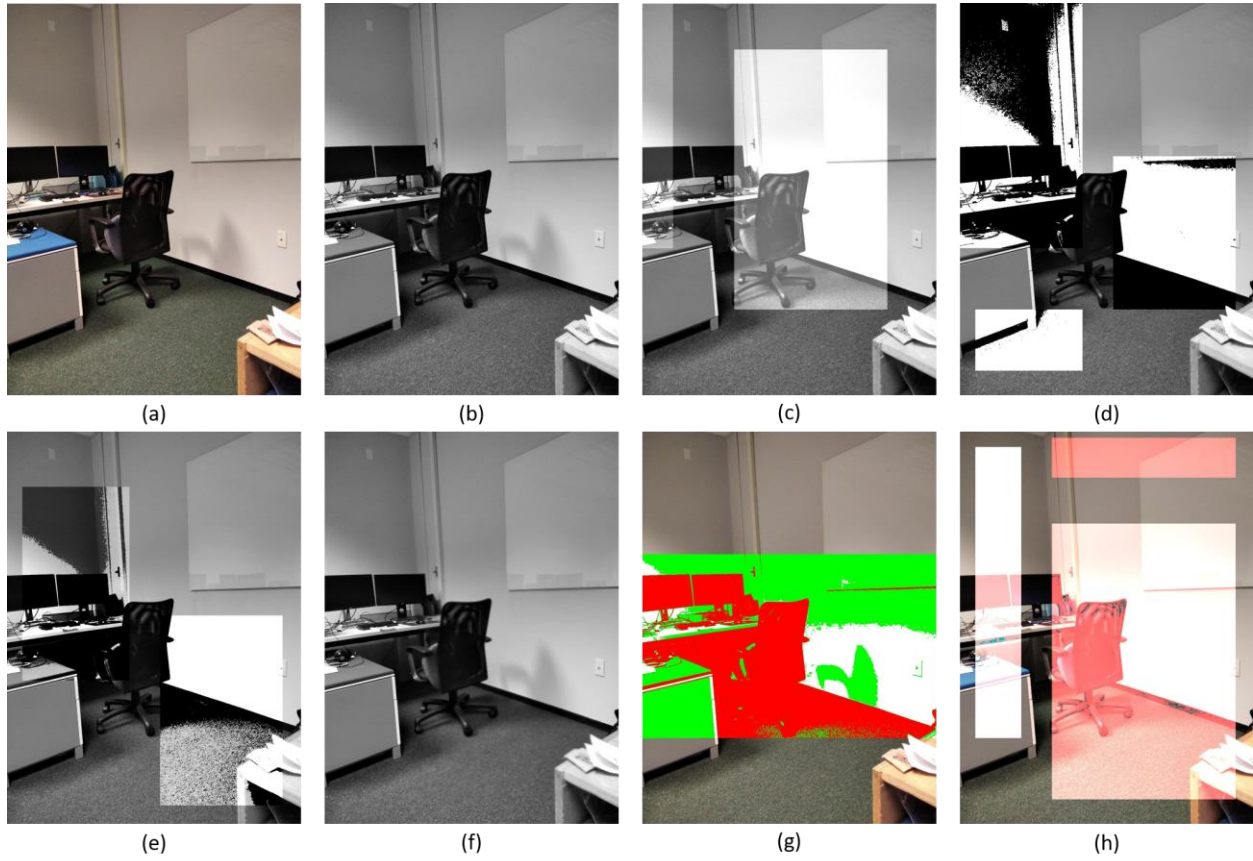    TC = 100 and C (20, 180, 20) for ROI2

# Result



*Figure 7 (a) Original image (b) Grayscale image (c) Add gray (d) Binarize
(e) Brightness (f) Smoothing (g) Color binarize (h) Color brightness*