**Assignment 1 Report**

**Dewi Kharismawati / 14231619**

For this assignment, I write 1 main script and 3 functions to fulfill all the requirement. The main purpose for this assignment is to assess the brightness and contrast of an image. We process an image to get their balance on the brightness and contrast. So that the image will be clearer.

The main file is in the assgn1\_Dewi. Inside that file, it going to call histogram (my custom histogram builder), histoCum (my custom histogram cumulative), and histoEqualization(my custom equalization function).

In the assign1\_Dewi file, you can uncomment images that you want to test.

%read the image

img = imread('Fig0310(b)(washed\_out\_pollen\_image).tif');

%================================================================

%in case you wanna try other images

%img = imread('Fig0312(a)(kidney).tif');

%================================================================

%img = imread('Fig0309(a)(washed\_out\_aerial\_image).tif');

%================================================================

%img = imread('Fig0323(a)(mars\_moon\_phobos).tif');

%================================================================

%img = imread('Fig0316(1)(top\_left).tif');

%================================================================

grayImage = img;

%display the image

imshow(img);

title('Original Image');

%in case the image is not grayscale, we convert them before processing the

%height = row

%width = column

%dept = the dept of the image

[height,width,depth]=size(img);

if depth > 1

grayImage = rgb2gray(img);

imshow(grayImage);

end

% to make the figure full screen

histPane = figure('units','normalized','outerposition',[0 0 1 1]);

%creating the histogram from the function that has been built called histo

%histo(image name, subplot position, title)

images\_histogram = histo(grayImage, 1,'Original Histogram');

%creating the original cummulative histogram

%histoCum(image\_histogram (from the histogram function), subplot position, title)

cummulativeHistogram = histoCum(images\_histogram, 2, 'Original Cummulative');

%%normalizing the histogram to get the qualization

%histoEqualization(cummulative, img, position, title);

S = histoEqualization(cummulativeHistogram, img, 3, 'Original Equalization');

%%after getting all the information from original images

% Assign the values or each gray value in the images

for i = 1:height

for j = 1:width

grayImage(i,j)=S(grayImage(i,j)+1);

end

end

%creating the histogram from the function that has been built called histo

%histo(image name, subplot position, title)

new\_images\_histogram = histo(grayImage, 4,'New Image Histogram');

%creating the original cummulative histogram

%histoCum(image\_histogram (from the histogram function), subplot position, title)

new\_cummulativeHistogram = histoCum(new\_images\_histogram, 5, 'New Image Cummulative');

%%normalizing the histogram to get the qualization

%histoEqualization(cummulative, img, position, title);

S = histoEqualization(new\_cummulativeHistogram, img, 6, 'New Image Equalization');

%display the new gray image in the figure 4

figure('units','normalized','outerposition',[0 0 1 1]);

imshow(grayImage);

title('New Image');

# Function 1 is histo.m

Histo is my custom histogram builder, I follow the tutorial in the slide. Histogram main function is to plots the number of pixels for each tonal value. Here is the code for histo.m

function f = histo(img, position, histoTitle)

%image is the input image

%position is where the histogram will be put on display

%histoTitle is the title of histogram on display

%dimention of the image

[height,width] = size(img);

%histogram initialization

%256 is the number of bins on the histogram

images\_histogram = zeros(1,256);

%building the histogram

for i = 1:height

for j = 1:width

images\_histogram(img(i,j)+1) = images\_histogram(img(i,j)+1)+1;

end

end

%draw the histogram on the diplay that been divided

subplot(2,3,position);

%drawing the bars

bar(images\_histogram);

%accessories for the title

title(histoTitle);

%return

f = images\_histogram;

end

Function 2 is the Histogram cumulative custom.

Here is the code for that histogram cumulative

function f = histoCum(images\_histogram, position, histCumTitle)

%histoCum is a function to build a cummulative histogram from an image

% images\_histogram: images histogram that is built from histo function

% position: position of the histogram in subplot

% histCumTitle: histogram title for the subplot

% Calculating the cummulative from bins histogram

cumulative = zeros(1,256);

for i = 1:256

if i == 1

cumulative(i) = images\_histogram(i);

else

cumulative(i) = cumulative(i-1) + images\_histogram (i);

end

end

%subplot to make the graph in a figure

subplot(2,3,position);

%puting the cummulative histogram on bin

bar(cumulative);

%titling

title(histCumTitle);

f = cumulative;

end

Function 3 is the Histogram Equalization custom for enhancing the image quality. The main purpose of this function is to enhancing the contrast of the image.

Here is the code for that histogram equalization.

function f = histoEqualization(cumulative,img,position,histEqTitle)

%histoEqualization a function to equalize and normalize the image histogram

% cumulative: histogram cummulative from histoCum

% img: the input image

% position: the graph position on the subplot

% histEqTitle: histogram title on the subplot

[image\_height,image\_width]=size(img);

L = 256;

% calculate the probability of the gray level

Pr = zeros(1,L);

for i = 1:L

Pr(i) = cumulative(i) / (image\_height \* image\_width);

S(i) = round((L-1)\*Pr(i));

end

%subplot to make the graph in a figure

subplot(2,3,position);

%putting on bars

bar(S);

%putting title

title(histEqTitle);

%return to S

f = S;

end

# RESULT

After we run the program, the result that we get is 3 figures.

Figure 1: Original Image

Figure 2: the comparison between original image histograms and new image histograms

Figure 3: new Image after the processing.

Figure 1

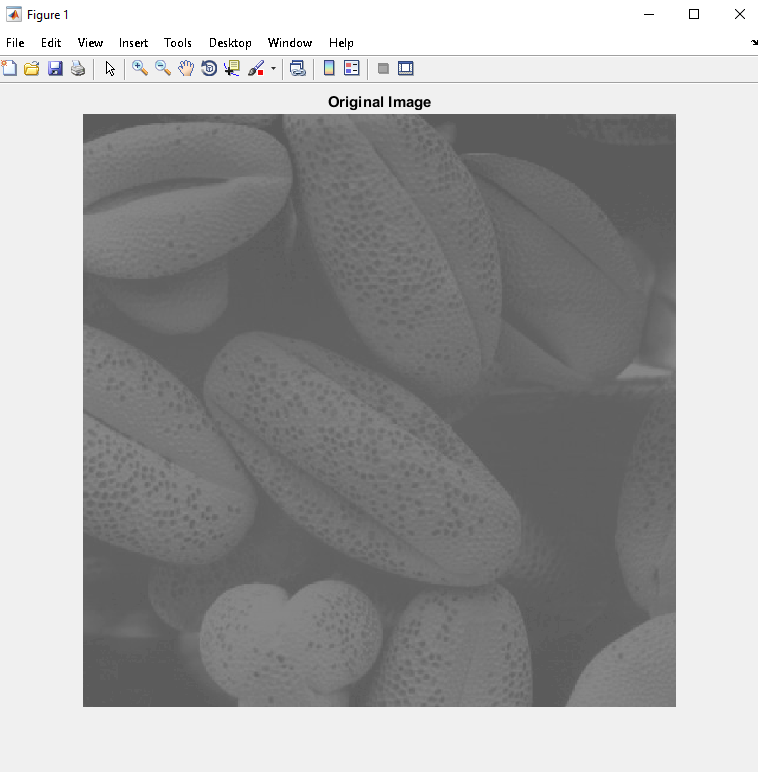


Figure 2 (this seems like there is a gap in the histogram), but when we zoom it in, the gap will be gone. It happens because I divided the figure into multiple subplots.

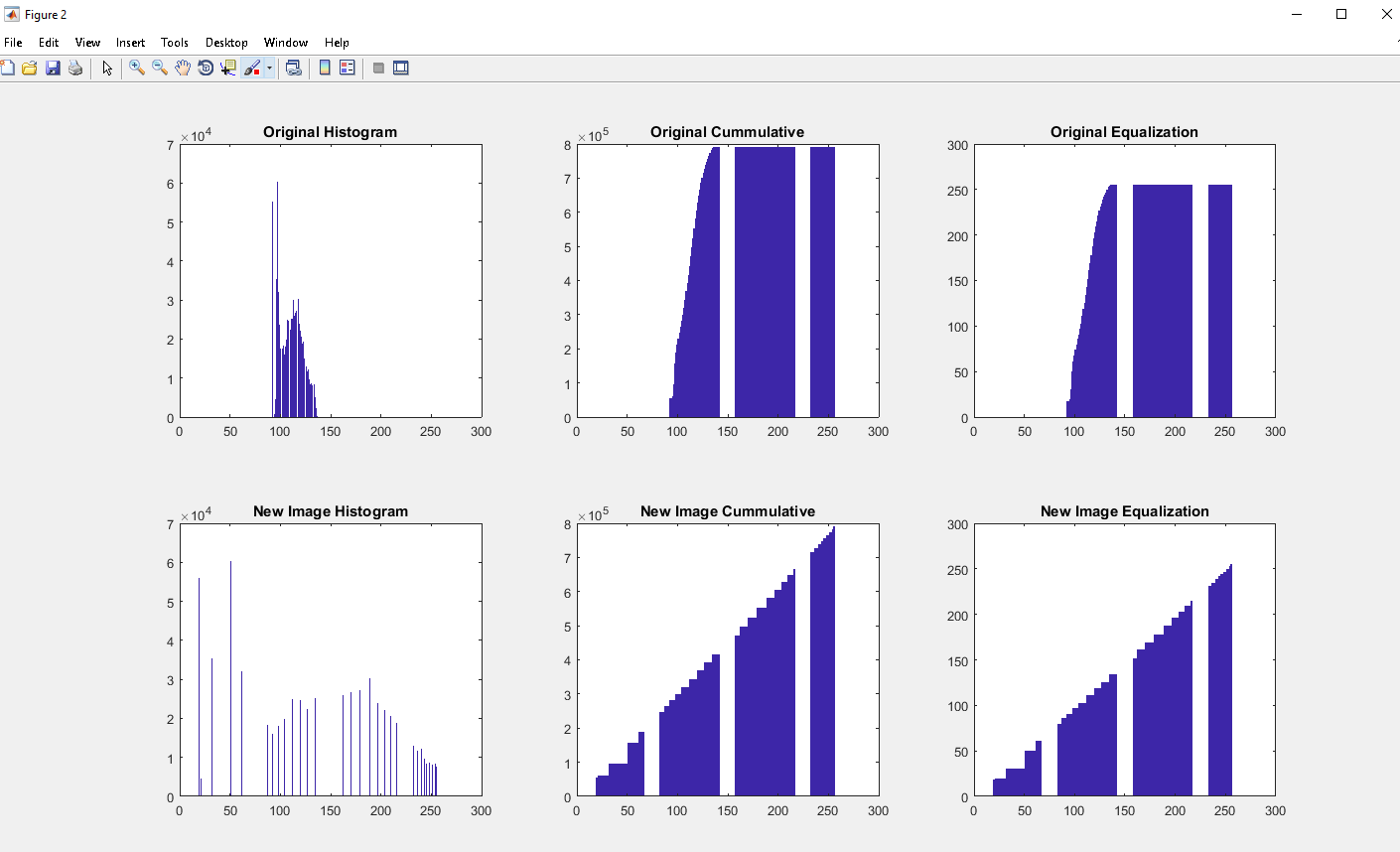


Figure 3

