

Lab 02

Submitted By:

Md. Aahadul Islam Fardin

ID: 2019-1-60-224

Section: 02

Course Code: CSE438(Image Processing)

Submitted To:

**Dr. Ahmed Wasif Reza**

Professor  
Department of Computer Science & Engineering

Date: 28-02-2023

**Problem No:1**

Code:

inputImg = imread('tree.jpg');

refImg = imread("tree\_reference.png");

if size(refImg, 3) == 3

refImg = rgb2gray(refImg);

end

inputHist = imhist(inputImg);

refHist = imhist(refImg);

inputCDF = cumsum(inputHist) / numel(inputImg);

refCDF = cumsum(refHist) / numel(refImg);

transFunc = interp1(refCDF, 0:255, inputCDF, 'linear');

outputImg = uint8(transFunc(double(inputImg) + 1));

subplot(1,2,1); imshow(inputImg); title('Input Image');

subplot(1,2,2); imshow(outputImg); title('Output Image');

Output:



Problem No: 2

Code:

inputImg = imread('CT.jpg')

outputImg = histeq(inputImg);

subplot(2,2,1); imshow(inputImg); title('Input Image');

subplot(2,2,2); imshow(outputImg); title('Output Image');

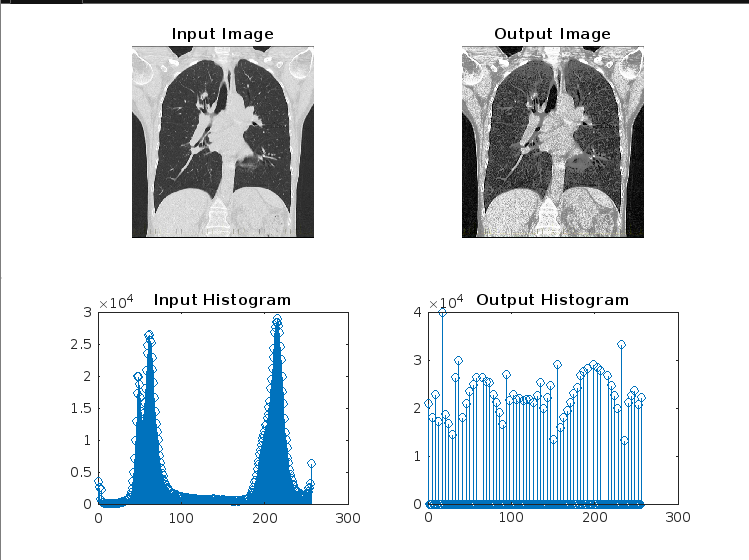
inputHist = imhist(inputImg);

outputHist = imhist(outputImg);

subplot(2,2,3); stem(inputHist); title('Input Histogram');

subplot(2,2,4); stem(outputHist); title('Output Histogram');

Output:



Problem No: 3

Code:

inputImg = imread('MRI.jpg');

noisyImg = imnoise(inputImg, 'salt & pepper', 0.05);

minFilteredImg = ordfilt2(noisyImg, 1, true(3,3));

maxFilteredImg = ordfilt2(noisyImg, 9, true(3,3));

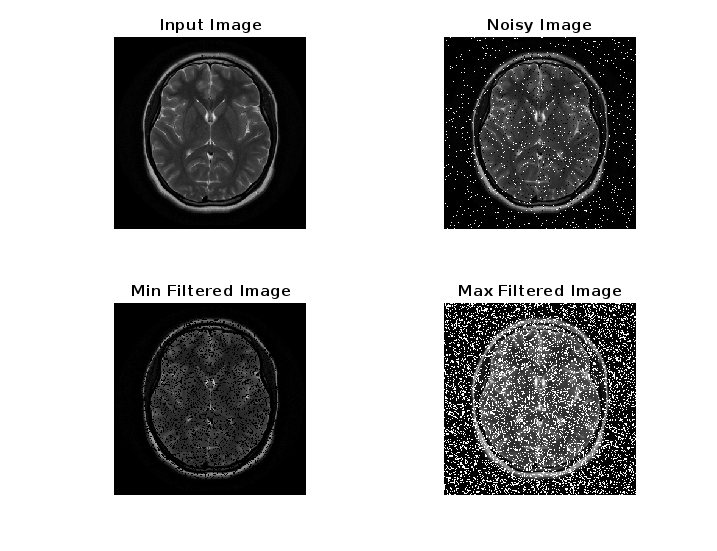
subplot(2,2,1); imshow(inputImg); title('Input Image');

subplot(2,2,2); imshow(noisyImg); title('Noisy Image');

subplot(2,2,3); imshow(minFilteredImg); title('Min Filtered Image');

subplot(2,2,4); imshow(maxFilteredImg); title('Max Filtered Image');

Output:



Problem No: 4

Code:

inputImg = imread('MRI.jpg');

noisyImg = imnoise(inputImg, 'gaussian', 0, 0.01);

minFilteredImg = ordfilt2(noisyImg, 1, true(3,3));

maxFilteredImg = ordfilt2(noisyImg, 9, true(3,3));

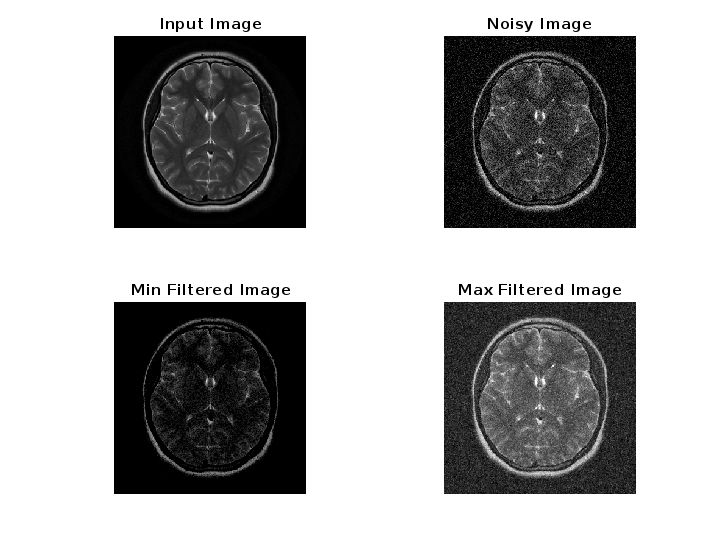
subplot(2,2,1); imshow(inputImg); title('Input Image');

subplot(2,2,2); imshow(noisyImg); title('Noisy Image');

subplot(2,2,3); imshow(minFilteredImg); title('Min Filtered Image');

subplot(2,2,4); imshow(maxFilteredImg); title('Max Filtered Image');

Output:



Problem No: 5

Code:

inputImg = imread('MRI\_2.jpg');

if size(inputImg, 3) == 3

inputImg = rgb2gray(inputImg);

end

noisyImg = imnoise(inputImg, 'gaussian', 0, 0.01);

boxFilteredImg = imboxfilt(noisyImg, 3);

averageFilteredImg = imfilter(noisyImg, fspecial('average', 3));

medianFilteredImg = medfilt2(noisyImg, [3, 3]);

figure;

subplot(2, 4, 1); imshow(inputImg); title('Input Image');

subplot(2, 4, 2); imshow(noisyImg); title('Noisy Image');

subplot(2, 4, 3); imshow(boxFilteredImg); title('Box Filtered Image');

subplot(2, 4, 4); imshow(averageFilteredImg); title('Average Filtered Image');

subplot(2, 4, 5); imshow(medianFilteredImg); title('Median Filtered Image');

boxPSNR = psnr(inputImg, boxFilteredImg);

averagePSNR = psnr(inputImg, averageFilteredImg);

medianPSNR = psnr(inputImg, medianFilteredImg);

disp(['Box Filtered PSNR: ' num2str(boxPSNR) ' dB']);

disp(['Average Filtered PSNR: ' num2str(averagePSNR) ' dB']);

disp(['Median Filtered PSNR: ' num2str(medianPSNR) ' dB']);

figure;

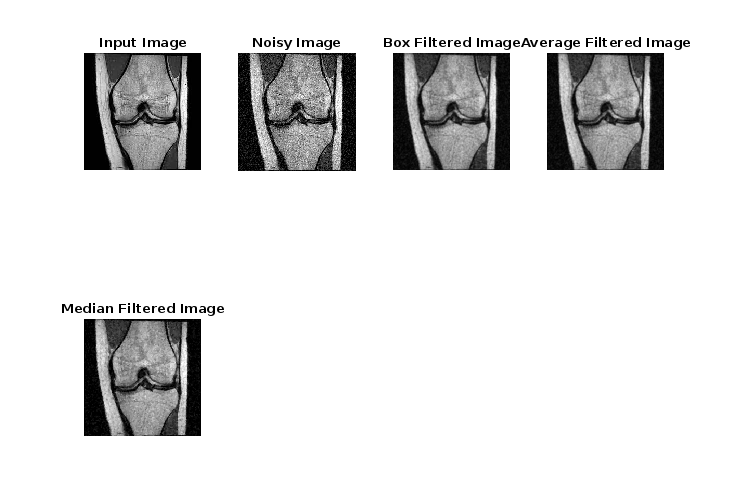
bar([boxPSNR, averagePSNR, medianPSNR]);

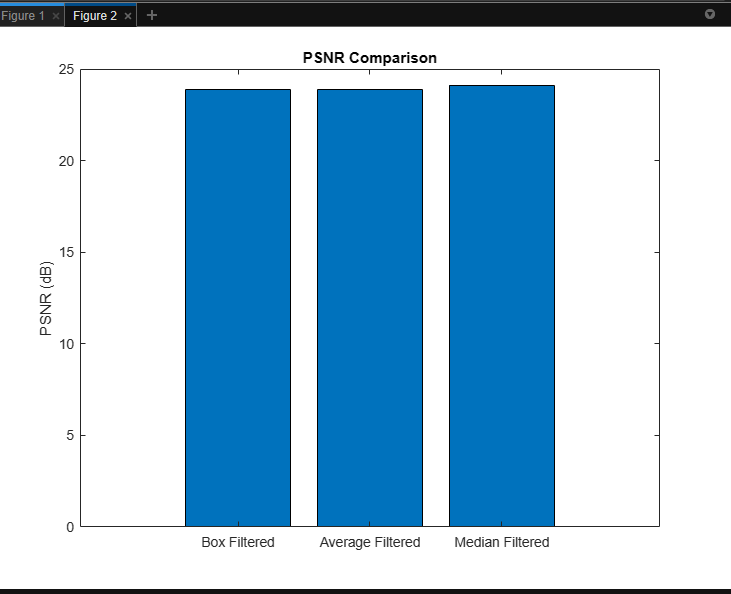
title('PSNR Comparison');

ylabel('PSNR (dB)');

xticklabels({'Box Filtered','Average Filtered','Median Filtered'});

Output:





Problem No: 6

Code:   
input\_img = imread('contrast.png');

output\_img = histeq(input\_img);

figure;

subplot(1, 2, 1);

imshow(input\_img);

title('Input Image');

subplot(1, 2, 2);

imshow(output\_img);

title('Output Image');

figure;

subplot(2, 1, 1);

imhist(input\_img);

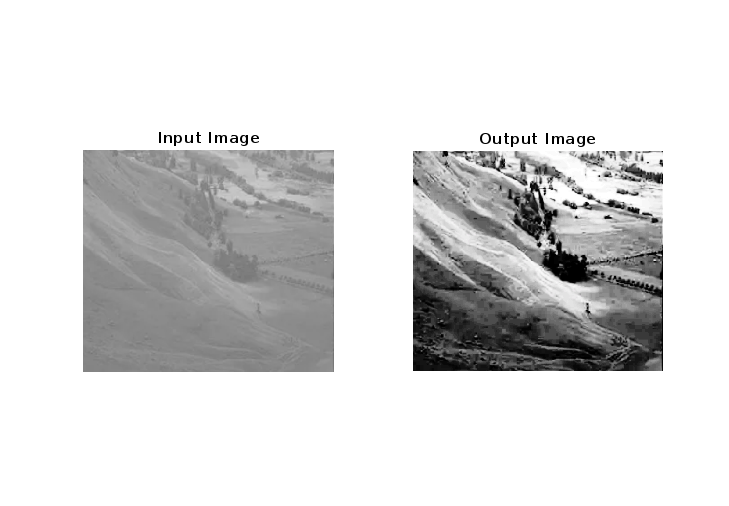
title('Input Image Histogram');

subplot(2, 1, 2);

imhist(output\_img);

title('Output Image Histogram');

Output:



Problem No: 7

Code:

input\_img = imread('Brightness.png');

brightness\_adj = 120;

output\_img = imadjust(input\_img, [], [], brightness\_adj/255);

figure;

subplot(1, 2, 1);

imshow(input\_img);

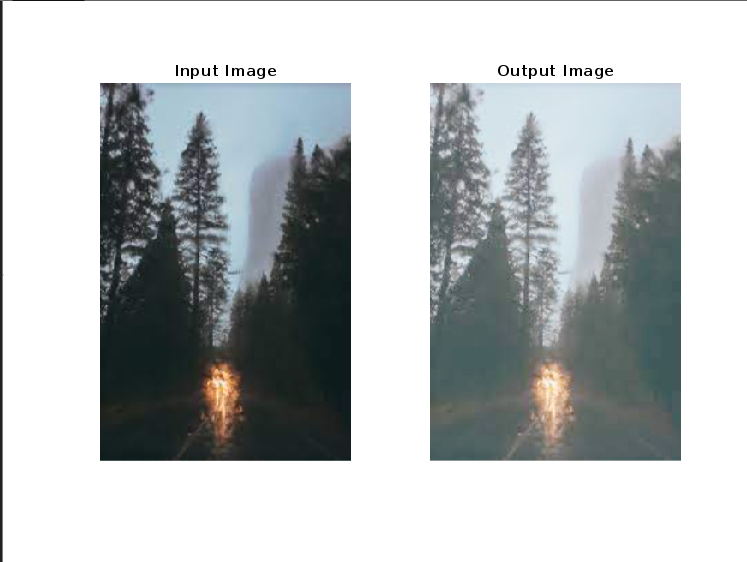
title('Input Image');

subplot(1, 2, 2);

imshow(output\_img);

title('Output Image');

Output:



Problem No: 8

Code:

input\_img = imread('coins.png');

output\_img = uint8(floor(double(input\_img)/32) \* 32);

figure;

subplot(1, 2, 1);

imshow(input\_img);

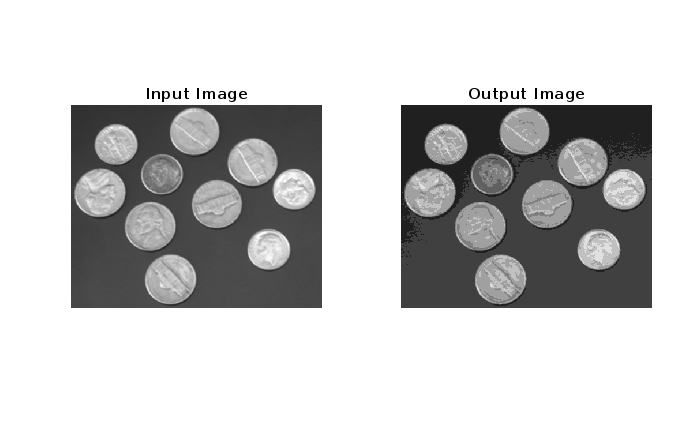
title('Input Image');

subplot(1, 2, 2);

imshow(output\_img);

title('Output Image');

Output:



Problem No: 9

Code:

input\_img = imread('Galaxy.png');

% show the input image

figure;

imshow(input\_img);

title('Input Image');

% show the matrix form of the image

disp('Image matrix:');

disp(input\_img);

% display pixel information by hovering cursor over image

imtool(input\_img);

% find the value of pixel (10,78)

pixel\_value = input\_img(10,78);

fprintf('Pixel value at (10,78): %d\n', pixel\_value);

% show the size of the image

img\_size = size(input\_img);

fprintf('Image size: %d x %d\n', img\_size(1), img\_size(2));

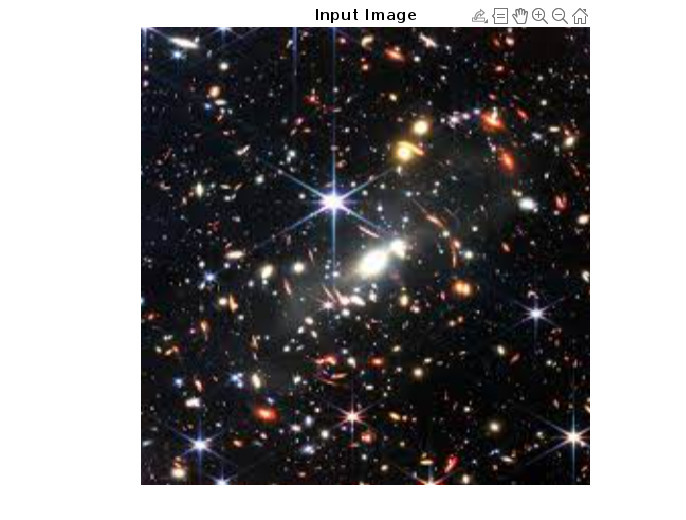
% show all image information

imfinfo('input\_image.jpg')

Output:

Pixel value at (10,78): 45

Image size: 433 x 425



Problem No:10

Code:

rgb\_img = imread('rgb\_image.png');

gray\_img = imread('grayscale\_\_image.png');

indexed\_img = imread('indexed\_\_image.png');

subplot(1,3,1);

imshow(rgb\_img);

title('RGB Image');

subplot(1,3,2);

imshow(gray\_img);

title('Grayscale Image');

subplot(1,3,3);

imshow(indexed\_img);

title('Indexed Image');

gray\_rgb\_img = rgb2gray(rgb\_img);

gray\_indexed\_img = ind2gray(indexed\_img, gray(256));

binary\_img = imbinarize(gray\_img);

inverted\_binary\_img = imcomplement(binary\_img);

figure;

imhist(gray\_img);

title('Histogram of Grayscale Image');

inverted\_rgb\_img = imcomplement(rgb\_img);

blurred\_rgb\_img = imgaussfilt(rgb\_img, 5);

Output:

