

**Digital Image Processing (CSE-438)**

**Sec:03**

**Lab: 03**

**Submitted to:**

Md. Asif Khan Rifat

Lecturer  
Department of Computer Science & Engineering

East West University

**Submitted by:**

Prinom Mojumder

2021-2-60-098

1. Apply salt and pepper noise to the following image and remove the noise using min and

max filtering technique. Show input and output side by side.

Code:

img = imread("Picture1.jpg");

if size(img,3) == 3

img = rgb2gray(img);

end

noisy\_img = imnoise(img, 'salt & pepper', 0.02);

se = ones(3,3);

min\_filtered\_img = ordfilt2(noisy\_img, 1, se);

max\_filtered\_img = ordfilt2(noisy\_img, 9, se);

figure;

subplot(2,2,1), imshow(img), title('Original Image');

subplot(2,2,2), imshow(noisy\_img), title('Noisy Image (Salt & Pepper)');

subplot(2,2,3), imshow(min\_filtered\_img), title('Min Filtered Image');

subplot(2,2,4), imshow(max\_filtered\_img), title('Max Filtered Image');

Output:

A group of images of a brain

AI-generated content may be incorrect.

1. Apply Gaussian noise to the following image and remove the noise using Gaussian

filtering. Show input and output side by side.

Code:

img = imread('Picture2.jpg');

if size(img,3) == 3

img = rgb2gray(img);

end

noisy\_img = imnoise(img, 'gaussian', 0, 0.01);

gaussian\_filter = fspecial('gaussian', [5 5], 1);

filtered\_img = imfilter(noisy\_img, gaussian\_filter, 'same');

figure;

subplot(1,3,1), imshow(img), title('Original Image');

subplot(1,3,2), imshow(noisy\_img), title('Gaussian Noisy Image');

subplot(1,3,3), imshow(filtered\_img), title('Denoised Image (Gaussian Filter)');

Output:

A close-up of a brain scan

AI-generated content may be incorrect.

1. Apply any noise to the following image and restore it using:

a) Box filtering

b) Average filtering

c) Median filtering

Show input and output side by side. Also show the comparison between the 3 techniques.

Mention which method works better than others.

Code:

img = imread('Picture3.jpg');

if size(img,3) == 3

img = rgb2gray(img);

end

noisy\_img = imnoise(img, 'salt & pepper', 0.02);

box\_filter = fspecial('average', [3 3]);

box\_filtered\_img = imfilter(noisy\_img, box\_filter, 'same');

average\_filter = fspecial('average', [5 5]);

average\_filtered\_img = imfilter(noisy\_img, average\_filter, 'same');

median\_filtered\_img = medfilt2(noisy\_img, [3 3]);

figure;

subplot(2,3,1), imshow(img), title('Original Image');

subplot(2,3,2), imshow(noisy\_img), title('Image with Noise');

subplot(2,3,3), imshow(box\_filtered\_img), title('Image with Box Filtering');

subplot(2,3,4), imshow(average\_filtered\_img), title('Image with Average Filtering');

subplot(2,3,6), imshow(median\_filtered\_img), title('Image with Median Filtering');

Output:

Several images of knee joint

AI-generated content may be incorrect.

1. Adjust the histogram of the following image to match the reference image using histogram matching. Show the histogram of original, reference, and output images.

Code:

% a) Read and show the image

img = imread('Picture4.jpg');

figure;

imshow(img);

title('Loaded Image');

% b) Show the matrix form of the image

disp('Matrix form of the image:');

disp(img);

% c) Show pixel information by hovering the cursor

impixelinfo;

% d) Find the value of the pixel (10, 78) - handles RGB now!

x = 10;

y = 78;

pixel\_value = img(x, y, :);

disp(['Pixel value at (10, 78): R=', num2str(pixel\_value(1)), ...

', G=', num2str(pixel\_value(2)), ', B=', num2str(pixel\_value(3))]);

% e) Show the size of the image

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

disp(['Image size: ', num2str(width), 'x', num2str(height), ', Channels: ', num2str(channels)]);

% f) Show all the information of the image

disp('Full image details:');

disp(['Dimensions: ', mat2str(size(img))]);

disp(['Data type: ', class(img)]);

disp(['Number of pixels: ', num2str(numel(img))]);

Output: (a,c)

A close-up of stars in space

AI-generated content may be incorrect.

Output: (b,d,e,f)

A white background with black numbers

AI-generated content may be incorrect.

1. Change the contrast of the image using histogram equalization. Show the histogram of both input and output images.

Code:

% Read the images

indexed\_img\_path = 'C:\Users\Students\Downloads\Lab 3\Picture5\_3.tif';

gray\_img\_path = 'C:\Users\Students\Downloads\Lab 3\Picture5\_2.jpg';

rgb\_img\_path = 'C:\Users\Students\Downloads\Lab 3\Picture5\_1.jpg';

% Read RGB and Grayscale Images

rgb\_img = imread(rgb\_img\_path);

gray\_img = imread(gray\_img\_path);

% Check if grayscale image is already single-channel

if size(gray\_img, 3) == 3

gray\_img = rgb2gray(gray\_img);

end

% Read Indexed Image & Check if It Has a Colormap

[indexed\_img, map] = imread(indexed\_img\_path);

if isempty(map)

indexed\_gray = gray\_img; % No colormap, assume grayscale

indexed\_rgb = rgb\_img; % Assume already RGB

else

indexed\_gray = ind2gray(indexed\_img, map);

indexed\_rgb = ind2rgb(indexed\_img, map);

end

% Convert RGB to Grayscale

rgb\_to\_gray = rgb2gray(rgb\_img);

% Convert Grayscale to Binary

binary\_img = imbinarize(gray\_img);

% Invert Binary Image

inverted\_binary = imcomplement(binary\_img);

% Histogram of Grayscale Image

figure, imhist(gray\_img), title('Histogram of Grayscale Image');

% Invert RGB Image

inverted\_rgb = imcomplement(im2uint8(rgb\_img)); % Convert to uint8 to avoid issues

% Blur RGB Image

blurred\_rgb = imgaussfilt(rgb\_img, 2);

% Display results

figure;

subplot(3,3,1), imshow(rgb\_img), title('RGB Image');

subplot(3,3,2), imshow(gray\_img), title('Grayscale Image');

subplot(3,3,3), imshow(indexed\_rgb), title('Indexed Image (Converted to RGB)');

subplot(3,3,4), imshow(rgb\_to\_gray), title('RGB to Grayscale');

subplot(3,3,5), imshow(indexed\_gray), title('Indexed to Grayscale');

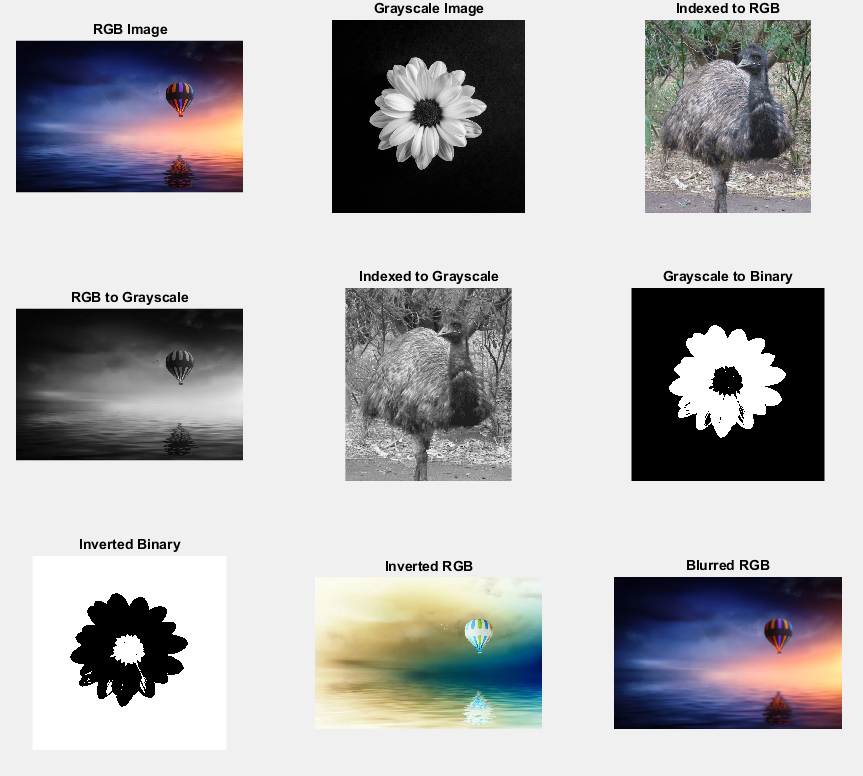
subplot(3,3,6), imshow(binary\_img), title('Binary Image');

subplot(3,3,7), imshow(inverted\_binary), title('Inverted Binary');

subplot(3,3,8), imshow(inverted\_rgb), title('Inverted RGB');

subplot(3,3,9), imshow(blurred\_rgb), title('Blurred RGB');

Output: (a,b,c,d,e,f,h,i)



Output: (g)

A graph of a graph

AI-generated content may be incorrect.