



प्रयोग सूची (List of Practicals)

सीएसई- तृतीय वर्ष / छठा सेमेस्टर (CSE- III Year / VI Semester)

सत्र: जनवरी-जून, 2025 (Session: Jan-June, 2025)

डिजिटल इमेज प्रोसेसिंग प्रयोगशाला (सीएसई-325) ,Digital Image Processing Lab (CSE-325)

लैब असाइनमेंट -1

Lab Assignment-1

a. Become familiar with MATLAB Basic commands / मैटलैब की बेसिक कमांड से परिचित हों

b. Read and display image in MATLAB / मैटलैब में छवि पढ़ें और प्रदर्शित करें

`imagePath = 'आपकी_छवि_फ़ाइल_का_पथ.jpg';` % Replace with your image file path / अपने छवि फ़ाइल के पथ से बदलें

```
imageData = imread(imagePath);
```

```
imshow(imageData);
```

```
title('Original Image');
```

c. Resize given image / दी गई छवि का आकार बदलें

`resizedImage = imresize(imageData, [300, 400]);` % Replace with desired dimensions / वांछित आयामों से बदलें

```
figure, imshow(resizedImage);
```

```
title('Resized Image');
```

d. Convert given color image into gray-scale image / दी गई रंगीन छवि को ग्रे-स्केल छवि में बदलें

```
grayImage = rgb2gray(imageData);
```

```
figure, imshow(grayImage);
```

```
title('Gray-Scale Image');
```



e. Convert given color/gray-scale image into black & white image / दी गई रंग/ग्रे-स्केल छवि को काले और सफेद छवि में बदलें

```
bwlImage = im2bw(imageData);  
figure, imshow(bwlImage);  
title('Black & White Image');
```

f. Draw image profile / छवि प्रोफ़ाइल बनाएं

```
profile = imageData(:, size(imageData, 2)/2, :); % Get a vertical profile from the middle / छवि के बीच से एक लंबी प्रोफ़ाइल प्राप्त करें  
figure, plot(profile);  
title('Image Profile');
```

g. Separate color image in three R, G & B planes / रंगीन छवि को तीन R, G और B प्लेन में अलग करें

```
redPlane = imageData(:, :, 1);  
greenPlane = imageData(:, :, 2);  
bluePlane = imageData(:, :, 3);
```

h. Create color image using R, G and B three separate planes / R, G, और B तीन अलग प्लेनों का उपयोग करके रंगीन छवि बनाएं

```
colorImage = cat(3, redPlane, greenPlane, bluePlane);  
figure, imshow(colorImage);  
title('Color Image using R, G & B planes');
```



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

लैब असाइनमेंट -२

Lab Assignment-2

बिंदु प्रसंस्करण विधि का उपयोग करके इमेज प्रोसेसिंग प्रोग्राम लिखें और निष्पादित करें।

Write and execute image processing programs using point processing method

a. Obtain Negative image / नकारात्मक छवि प्राप्त करें

imagePath = 'आपकी_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your image file path / अपने छवि फ़ाइल के पथ से बदलें

```
originalImage = imread(imagePath);
```

```
negativeImage = 255 - originalImage; % Invert pixel values / पिक्सल मानों को उलटें
```

```
imshow(negativeImage);
```

```
title('Negative Image');
```

b. Obtain Flip image / फ़्लिप छवि प्राप्त करें

```
flipImage = flipud(originalImage); % Flip the image vertically / छवि को ऊपर से नीचे की ओर फ़्लिप करें
```

```
imshow(flipImage);
```

```
title('Flip Image');
```

c. Perform Thresholding / थ्रेशोल्डिंग पेरफ़ॉर्म करें

```
thresholdValue = 128; % Adjust threshold value as needed / आवश्यकतानुसार थ्रेशोल्ड मान को समायोजित करें
```

```
thresholdedImage = originalImage > thresholdValue; % Convert to binary image / बाइनरी छवि में परिवर्तित करें
```

```
imshow(thresholdedImage);
```



```
title('Thresholded Image ');
```

d. Perform contrast stretching / कॉन्ट्रास्ट स्ट्रेचिंग करें

```
minPixelValue = min(originalImage(:));
```

```
maxPixelValue = max(originalImage(:));
```

```
desiredMin = 0;
```

```
desiredMax = 255;
```

```
contrastStretchedImage = (originalImage - minPixelValue) * (desiredMax - desiredMin) / (maxPixelValue - minPixelValue) + desiredMin;
```

```
contrastStretchedImage = uint8(contrastStretchedImage); % Convert to uint8 / uint8 में परिवर्तित करें
```

```
imshow(contrastStretchedImage);
```

```
title('Contrast Stretched Image ');
```

लैब असाइनमेंट ३

Lab Assignment 3

छवि अंकगणितीय प्रक्रियाओं के लिए कार्यक्रम लिखें और निष्पादित करें।

Write and execute programs for image arithmetic operations

a. Addition of two images / दो छवियों का जोड़।

imagePath1 = 'आपकी_पहली_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your first image file path / अपनी पहली छवि फ़ाइल के पथ से बदलें

imagePath2 = 'आपकी_दूसरी_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your second image file path / अपनी दूसरी छवि फ़ाइल के पथ से बदलें

```
image1 = imread(imagePath1);
```

```
image2 = imread(imagePath2);
```



```
sumImage = image1 + image2; % Perform element-wise addition
```

```
imshow(sumImage);
```

```
title('Sum of Images ');
```

b. Subtract one image from the other image / एक छवि को दूसरी छवि से कम करें।

```
subtractedImage = image1 - image2; % Perform element-wise subtraction
```

```
imshow(subtractedImage);
```

```
title('Subtracted Image ');
```

c. Calculate the mean value of the image / छवि का औसत मूल्य निर्गण करें।

```
meanValue = mean2(image1); % Calculate mean value
```

```
disp(['Mean Value of Image: ', num2str(meanValue)]);
```

d. Vary brightness by changing the mean value / मान मूल्य बदलकर अलग उज्ज्वलता कीजिए।

```
desiredMean = 150; % Adjust desired mean value
```

```
adjustedImage = image1 + (desiredMean - meanValue); % Change brightness by adjusting pixel values
```

```
imshow(adjustedImage);
```

```
title('Brightness Adjusted Image ');
```

लैब असाइनमेंट ४

Lab Assignment 4

छवि तार्किक प्रक्रियाओं के लिए प्रोग्राम लिखें और निष्पादित करें।

To write and execute programs for image logical operations



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

a. AND operation between two images / दो छवियों के बीच AND ऑपरेशन।

imagePath1 = 'आपकी_पहली_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your first image file path / अपनी पहली छवि फ़ाइल के पथ से बदलें

imagePath2 = 'आपकी_दूसरी_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your second image file path / अपनी दूसरी छवि फ़ाइल के पथ से बदलें

```
image1 = imread(imagePath1);
```

```
image2 = imread(imagePath2);
```

```
andImage = bitand(image1, image2); % Perform bitwise AND operation
```

```
imshow(andImage);
```

```
title('AND Operation Image ');
```

b. OR operation between two images / दो छवियों के बीच OR ऑपरेशन।

```
orImage = bitor(image1, image2); % Perform bitwise OR operation
```

```
imshow(orImage);
```

```
title('OR Operation Image ');
```

c. Calculate the intersection of two images / दो छवियों के सर्वनिष्ठ की गणना करें।

```
intersectionImage = bitand(image1, image2); % Calculate intersection using bitwise AND
```

```
imshow(intersectionImage);
```

```
title('Intersection Image ');
```

d. Water Marking using EX-OR operation / EX-OR ऑपरेशन का उपयोग करके वॉटरमार्किंग करें।

```
watermarkedImage = bitxor(image1, image2); % Perform EX-OR operation for watermarking
```

```
imshow(watermarkedImage);
```

```
title('Watermarked Image ');
```



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

e. NOT operation (Negative image)

```
notImage = bitcmp(image1); % Perform bitwise NOT operation
```

```
imshow(notImage);
```

```
title('NOT Operation Image ');
```

लैब असाइनमेंट ५

Lab Assignment 5

a. Program without using standard MATLAB functions and without standard MATLAB function.

प्रोग्राम बिना मानक MATLAB फ़ंक्शन और मानक MATLAB फ़ंक्शन का उपयोग किए बिना।

imagePath = 'आपकी_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your image file path / अपनी छवि फ़ाइल के पथ से बदलें

```
imageData = imread(imagePath);
```

```
grayImage = rgb2gray(imageData);
```

```
histogram = zeros(256, 1); % Initialize histogram array
```

```
[M, N] = size(grayImage); % Get dimensions of the image
```

```
for i = 1:M
```

```
    for j = 1:N
```

```
        pixelValue = grayImage(i, j);
```

```
        histogram(pixelValue + 1) = histogram(pixelValue + 1) + 1; % Increment histogram count
```

```
    end
```



end

b. Perform histogram equalization more than one times and compare the results.

हिस्टोग्राम समानता को एक से अधिक बार कार्यान्वित करें और परिणामों की तुलना करें।

```
numEqualizations = 3; % Number of equalization iterations
```

```
equalizedImages = cell(numEqualizations, 1);
```

```
for k = 1:numEqualizations
```

```
    equalizedImage = zeros(size(grayImage));
```

```
    for i = 1:M
```

```
        for j = 1:N
```

```
            pixelValue = grayImage(i, j);
```

```
            equalizedImage(i, j) = sum(histogram(1:pixelValue + 1)) / (M * N) * 255; % Equalization formula
```

```
        end
```

```
    end
```

```
    equalizedImages{k} = uint8(equalizedImage);
```

```
    histogram = hist(double(equalizedImage(:)), 256); % Update histogram for next iteration
```

```
end
```

c. Perform adaptive histogram equalization method.

आदर्श हिस्टोग्राम समानता विधि का कार्यान्वयन करें।

```
adapthisteqlImage = adapthisteq(grayImage);
```

d. Execute the local statistics histogram equalization.

स्थानीय आंकड़ी हिस्टोग्राम समानता को निष्पादित करें।

```
localstatImage = histeq(grayImage);
```



e. Compare all the above images and write comments on it.

उपरोक्त सभी छवियों की तुलना करें और उस पर टिप्पणियाँ लिखें।

figure;

subplot(2, 3, 1), imshow(grayImage), title('Original Image / मूल छवि');

for k = 1:numEqualizations

subplot(2, 3, k + 1), imshow(equalizedImages{k}), title(['Equalized #', num2str(k), ' / समान बनाई गई #', num2str(k)]);

end

subplot(2, 3, 5), imshow(adapthisteqImage), title('Adaptive Histeq / आदर्श हिस्टोग्राम समानता');

subplot(2, 3, 6), imshow(localstatImage), title('Local Stats Histeq / स्थानीय आंकड़ी हिस्टोग्राम समानता');

लैब असाइनमेंट ६

Lab Assignment 6

एक छवि पर विचार करें

Consider an image .

a. उस छवि में यादृच्छिक रूप से 20 बिंदु p और q का चयन करें और विभिन्न प्रकार की दूरी की गणना करें।

Select 20 point p and q randomly in that image and calculate the various type distance.

b. कुछ बिंदु p और q चुनें जहाँ p और q की पिक्सेल तीव्रता समान हो। p और q के बीच विभिन्न दूरी की गणना करें।

Select some point p and q where pixel intensity of p and q are same. Calculate the various distance between p and q .

c. पिक्सेल p और q की संख्या भी ज्ञात करें, जिनकी दूरी समान है।

Also find out the number of pixel p and q ,which have the same distance

d. मानों के समूह के बीच सबसे छोटी दूरी और सबसे लंबी दूरी की संख्या ज्ञात कीजिए।

Find the number of shortest distance and longest distance between the set of values.



a. Load and preprocess the image / छवि लोड करें और पूर्व-प्रसंस्करण करें

```
image = imread('your_image_path.jpg');  
grayImage = rgb2gray(image);
```

b. Number of random points / यादृच्छिक बिंदुओं की संख्या

```
numRandomPoints = 20;
```

c. Generate random points within the image / छवि के भीतर यादृच्छिक बिंदु उत्पन्न करें

```
imageSize = size(grayImage);  
randomPointsP = randi([1, imageSize(1)], numRandomPoints, 2);  
randomPointsQ = randi([1, imageSize(1)], numRandomPoints, 2);
```

d. Calculate distances between random points / यादृच्छिक बिंदुओं के बीच दूरी की गणना करें

```
euclideanDistances = sqrt(sum((randomPointsP - randomPointsQ).^2, 2));  
manhattanDistances = sum(abs(randomPointsP - randomPointsQ), 2);  
chebyshevDistances = max(abs(randomPointsP - randomPointsQ), [], 2);
```

e. Count occurrences of same distances / समान दूरियों की परिपत्र की गिनती करें

```
sameDistances = sum(euclideanDistances == manhattanDistances & manhattanDistances ==  
chebyshevDistances);
```

f. Display the results / परिणाम दिखाएँ

```
disp(['Number of occurrences with same distances: ', num2str(sameDistances)]);
```



लैब असाइनमेंट ७

Lab Assignment 7

एक RGB छवि पर विचार करें.

Consider an RGB color image.

a. 4 कनेक्टिविटी और 8-कनेक्टिविटी और सबसे छोटा रास्ता खोजें।

Find out 4 connectivity and 8-connectivity and shortest path.

b. इसे ग्रे स्केल में बदलें और पिक्सेल p और q के बीच विभिन्न दूरी की गणना करें।

convert it to gray scale and the calculate the various distance between the pixel p and q.

c. RGB प्लेन में पिक्सेल p और q के बीच की दूरी की भी गणना करें, व्यक्तिगत रूप से ग्रे स्केल में स्थित समान समन्वय पर विचार करें।

Also calculate the distance between pixel p and q in RGB plane individually consider the same coordinate located in Gray scale.

a. Load and preprocess the image / छवि लोड करें और पूर्व-प्रसंस्करण करें

```
image = imread('your_image_path.jpg');
```

```
grayImage = rgb2gray(image);
```

b. Define start and end points / प्रारंभ और समापन बिंदुओं की परिभाषा करें

```
start = [100, 100]; % Replace with your desired start point
```

```
endPoint = [200, 200]; % Replace with your desired end point
```

c. Implement 4-connectivity algorithm / 4-संयोजन एल्गोरिदम को कार्यान्वित करें

```
fourConnectivityNeighbors = [
```

```
0, -1; % Up
```

```
-1, 0; % Left
```



```
0, 1; % Down
```

```
1, 0 % Right
```

```
];
```

d. Implement 8-connectivity algorithm / 8-संयोजन एल्गोरिदम को कार्यान्वित करें

```
eightConnectivityNeighbors = [
```

```
0, -1; % Up
```

```
-1, -1; % Up-left
```

```
-1, 0; % Left
```

```
-1, 1; % Down-left
```

```
0, 1; % Down
```

```
1, 1; % Down-right
```

```
1, 0; % Right
```

```
1, -1 % Up-right
```

```
];
```

e. Create a graph for path finding / पथ खोज के लिए ग्राफ बनाएं

```
graph = zeros(size(grayImage));
```

f. Convert image to binary (0 and 1) / छवि को द्विआधिक (0 और 1) में रूपांतरित करें

```
binaryImage = grayImage > 128;
```

g. Assign values to graph based on image / छवि के आधार पर ग्राफ को मान दें

```
graph(binaryImage) = 1;
```



h. Implement Dijkstra's algorithm to find the shortest path / सबसे छोटे मार्ग को ढूंढने के लिए डायक्स्ट्रा का एल्गोरिदम कार्यान्वित करें

i. Load and preprocess the image / छवि लोड करें और पूर्व-प्रसंस्करण करें

```
image = imread('your_image_path.jpg');
```

```
grayImage = rgb2gray(image);
```

j. Define start and end points / प्रारंभ और समापन बिंदुओं की परिभाषा करें

```
start = [100, 100]; % Replace with your desired start point
```

```
endPoint = [200, 200]; % Replace with your desired end point
```

k. Create a binary image (0s and 1s) from the gray image / ग्रे छवि से द्विआधिक छवि (0 और 1) बनाएं

```
binaryImage = grayImage > 128;
```

l. Calculate the number of rows and columns in the image / छवि में पंक्तियों और स्तंभों की संख्या की गणना करें

```
[rows, cols] = size(binaryImage);
```

m. Convert 2D coordinates to 1D index / 2D संयोजनों को 1D सूची में रूपांतरित करें

```
coord2index = @(r, c) (r - 1) * cols + c;
```

n. Initialize distances and visited arrays / दूरियों और आवश्यक एरे की शुरुआत करें

```
distances = inf(rows, cols);
```

```
visited = false(rows, cols);
```

o. Set distance of start point to 0 / प्रारंभ बिंदु की दूरी को 0 पर सेट करें



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

```
startIdx = coord2index(start(1), start(2));
```

```
distances(start(1), start(2)) = 0;
```

p. Implement Dijkstra's algorithm / डायक्स्ट्रा का एल्गोरिदम कार्यान्वित करें

```
while true
```

```
q.
```

```
[minDistance, minIdx] = min(distances(:));
```

```
if visited(minIdx) || isinf(minDistance)
```

```
    break;
```

```
end
```

% Mark the current point as visited / वर्तमान बिंदु को देखा गया चिह्नित करें

```
visited(minIdx) = true;
```

% Convert 1D index back to 2D coordinates / 1D सूची को 2D संयोजनों में रूपांतरित करें

```
[currentRow, currentCol] = ind2sub([rows, cols], minIdx);
```

% Calculate the neighbors of the current point / वर्तमान बिंदु के पड़ोसियों की गणना करें

```
neighbors = [
```

```
    currentRow - 1, currentCol;
```

```
    currentRow + 1, currentCol;
```

```
    currentRow, currentCol - 1;
```

```
    currentRow, currentCol + 1;
```

```
];
```



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

```
for neighborIdx = 1:size(neighbors, 1)
    neighborRow = neighbors(neighborIdx, 1);
    neighborCol = neighbors(neighborIdx, 2);

    % Check if the neighbor is within bounds and unvisited / पड़ोसी सीमा के भीतर और अनदेखा है या नहीं देखें
    if neighborRow >= 1 && neighborRow <= rows && neighborCol >= 1 && neighborCol <= cols &&
        ~visited(neighborRow, neighborCol)

        % Calculate the distance to the neighbor / पड़ोसी के लिए दूरी की गणना करें
        neighborDistance = distances(currentRow, currentCol) + 1; % Assuming uniform cost for simplicity

        % Update distance if it's smaller / यदि यह छोटा है तो दूरी को अद्यतन करें
        if neighborDistance < distances(neighborRow, neighborCol)
            distances(neighborRow, neighborCol) = neighborDistance;
        end
    end
end
end

% Trace back the shortest path from end point to start point / समापन बिंदु से प्रारंभ बिंदु तक के सबसे छोटे
मार्ग को पारित करें
path = [];
currentRow = endPoint(1);
currentCol = endPoint(2);

while true
    path = [currentRow, currentCol; path];
```



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

```
if isequal([currentRow, currentCol], start)
    break;
end

neighbors = [
    currentRow - 1, currentCol;
    currentRow + 1, currentCol;
    currentRow, currentCol - 1;
    currentRow, currentCol + 1;
];

for neighborIdx = 1:size(neighbors, 1)
    neighborRow = neighbors(neighborIdx, 1);
    neighborCol = neighbors(neighborIdx, 2);

    if neighborRow >= 1 && neighborRow <= rows && neighborCol >= 1 && neighborCol <= cols
        if distances(neighborRow, neighborCol) == distances(currentRow, currentCol) - 1
            currentRow = neighborRow;
            currentCol = neighborCol;
            break;
        end
    end
end
end
```



% Display the result / परिणाम दिखाएँ

imshow(image);

hold on;

plot(path(:, 2), path(:, 1), 'r', 'LineWidth', 2);

title('Shortest Path between Start and End Points / प्रारंभ और समापन बिंदुओं के बीच सबसे छोटा मार्ग');

hold off;

लैब असाइनमेंट ८

Lab Assignment 8

स्थानिक फ़िल्टर का उपयोग करके शोर को निकालने के लिए प्रोग्राम लिखें और निष्पादित करें।

Write and execute programs to remove noise using spatial filters

% Load the noisy image / शोरयुक्त छवि लोड करें

imagePath = 'आपकी_शोरयुक्त_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your noisy image file path / अपनी शोरयुक्त छवि फ़ाइल के पथ से बदलें

noisyImage = imread(imagePath);

% Convert image to grayscale if needed / आवश्यकता होने पर छवि को ग्रेस्केल में रूपांतरित करें

if size(noisyImage, 3) == 3

noisyImage = rgb2gray(noisyImage);

end

% a. Understand 1-D and 2-D convolution process / 1-डी और 2-डी समांतरण प्रक्रिया को समझें

mask1D = [1, 2, 1]; % 1-D convolution mask / 1-डी समांतरण मास्क



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

convolvedImage1D = conv2(double(noisyImage), mask1D, 'same'); % Perform 1-D convolution / 1-डी समांतरण
कार्यान्वित करें

mask2D = [1, 2, 1; 2, 4, 2; 1, 2, 1]; % 2-D convolution mask / 2-डी समांतरण मास्क

convolvedImage2D = conv2(double(noisyImage), mask2D, 'same'); % Perform 2-D convolution / 2-डी समांतरण
कार्यान्वित करें

% Display the results / परिणाम दिखाएँ

figure;

subplot(2, 2, 1), imshow(noisyImage), title('Noisy Image / शोरयुक्त छवि');

subplot(2, 2, 2), imshow(convolvedImage1D, []), title('1-D Convolution / 1-डी समांतरण');

subplot(2, 2, 3), imshow(convolvedImage2D, []), title('2-D Convolution / 2-डी समांतरण');

% b. Use 3x3 Mask for low pass filter and high pass filter / 3x3 मास्क का उपयोग करके लो पास फ़िल्टर और
हाई पास फ़िल्टर के लिए

lowPassMask = ones(3) / 9; % 3x3 mask for low pass filter / लो पास फ़िल्टर के लिए 3x3 मास्क

lowPassFiltered = conv2(double(noisyImage), lowPassMask, 'same'); % Low pass filtering / लो पास फ़िल्टरिंग

highPassMask = [0, -1, 0; -1, 5, -1; 0, -1, 0]; % 3x3 mask for high pass filter / हाई पास फ़िल्टर के लिए 3x3 मास्क

highPassFiltered = conv2(double(noisyImage), highPassMask, 'same'); % High pass filtering / हाई पास फ़िल्टरिंग

% Display the filtered images / फ़िल्टरिंग की गई छवियों को दिखाएँ

figure;



```
subplot(1, 2, 1), imshow(lowPassFiltered, []), title('Low Pass Filtered / लो पास फ़िल्टरिंग');  
subplot(1, 2, 2), imshow(highPassFiltered, []), title('High Pass Filtered / हाई पास फ़िल्टरिंग');
```

लैब असाइनमेंट ९

Lab Assignment 9

तीव्रता स्तर स्लाइसिंग और बिट लेवल स्लाइसिंग को दर्शाने के लिए एक जीयूआई प्रोग्राम लिखें।

Write a GUI program, to illustrate intensity level slicing and bit level slicing.

% Create a GUI window / GUI विंडो बनाएं

```
gui = figure('Name', 'Intensity Level Slicing', 'Position', [100, 100, 400, 300]);
```

% Add input fields and buttons / इनपुट फ़िल्ड्स और बटन जोड़ें

```
uicontrol('Style', 'text', 'Position', [20, 220, 100, 25], 'String', 'Image Path:');
```

```
imagePathField = uicontrol('Style', 'edit', 'Position', [130, 220, 200, 25]);
```

```
uicontrol('Style', 'text', 'Position', [20, 170, 100, 20], 'String', 'Intensity Range:');
```

```
intensityRangeField = uicontrol('Style', 'edit', 'Position', [130, 170, 50, 20]);
```

```
applyButton = uicontrol('Style', 'pushbutton', 'Position', [300, 170, 80, 20], 'String', 'Apply Slicing');
```

% Define callback for the "Apply Slicing" button / "स्लाइसिंग लागू करें" बटन के लिए कॉलबैक निर्धारित करें

```
set(applyButton, 'Callback', @applySlicing);
```

% Define callback function for the button / बटन के लिए कॉलबैक फ़ंक्शन निर्धारित करें

```
function applySlicing(~, ~)
```

% Get input values / इनपुट मान प्राप्त करें



```
imagePath = get(imagePathField, 'String');  
intensityRange = str2double(get(intensityRangeField, 'String'));  
  
% Load the image / छवि लोड करें  
originalImage = imread(imagePath);  
  
% Convert to grayscale / ग्रेस्केल में रूपांतरित करें  
grayImage = rgb2gray(originalImage);  
  
% Create binary mask for slicing / स्लाइसिंग के लिए बाइनरी मास्क बनाएं  
mask = (grayImage >= intensityRange(1)) & (grayImage <= intensityRange(2));  
  
% Apply slicing / स्लाइसिंग कार्यान्वित करें  
slicedImage = grayImage;  
slicedImage(~mask) = 0;  
  
% Display images / छवियाँ दिखाएँ  
subplot(1, 2, 1), imshow(grayImage), title('Original Image / मूल छवि');  
subplot(1, 2, 2), imshow(slicedImage), title('Sliced Image / स्लाइस की गई छवि');  
end
```



Write a GUI program to illustrate Steganography.

% Create a GUI window / GUI विंडो बनाएं

```
gui = figure('Name', 'Steganography', 'Position', [100, 100, 400, 300]);
```

% Add input fields and buttons / इनपुट फ़िल्ड्स और बटन जोड़ें

```
uicontrol('Style', 'text', 'Position', [20, 220, 100, 25], 'String', 'Cover Image:');
```

```
coverImagePathField = uicontrol('Style', 'edit', 'Position', [130, 220, 200, 25]);
```

```
uicontrol('Style', 'text', 'Position', [20, 170, 100, 25], 'String', 'Secret Message:');
```

```
secretMessageField = uicontrol('Style', 'edit', 'Position', [130, 170, 200, 25]);
```

```
hideButton = uicontrol('Style', 'pushbutton', 'Position', [300, 130, 80, 30], 'String', 'Hide Message');
```

```
revealButton = uicontrol('Style', 'pushbutton', 'Position', [300, 80, 80, 30], 'String', 'Reveal Message');
```

% Define callbacks for buttons / बटनों के लिए कॉलबैक निर्धारित करें

```
set(hideButton, 'Callback', @hideMessage);
```

```
set(revealButton, 'Callback', @revealMessage);
```

% Define callback functions / कॉलबैक फ़ंक्शन निर्धारित करें

```
function hideMessage(~, ~)
```

% Get input values / इनपुट मान प्राप्त करें

```
coverImagePath = get(coverImagePathField, 'String');
```

```
secretMessage = get(secretMessageField, 'String');
```

% Load the cover image / कवर छवि लोड करें



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

```
coverImage = imread(coverImagePath);

% Convert the secret message to binary / गुप्त संदेश को बाइनरी में रूपांतरित करें
binaryMessage = str2bin(secretMessage);

% Embed the message using LSB substitution / LSB प्रतिस्थापन का उपयोग करके संदेश दबाएं
stegoImage = embedMessage(coverImage, binaryMessage);

% Display the stego image / स्टेगो छवि दिखाएँ
imshow(stegoImage);
title('Stego Image / स्टेगो छवि');
end

function revealMessage(~, ~)

% Load the stego image / स्टेगो छवि लोड करें
stegoImage = imread(get(coverImagePathField, 'String'));

% Extract the hidden message using LSB extraction / LSB निष्कर्षण का उपयोग करके छिपे हुए संदेश का निष्कर्षण करें
extractedMessage = extractMessage(stegoImage);

% Display the extracted message / निष्कर्षित संदेश दिखाएँ
disp(['Extracted Message: ', extractedMessage]);
end
```



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

% Function to convert string to binary / स्ट्रिंग को बाइनरी में रूपांतरित करने के लिए फ़ंक्शन

```
function binary = str2bin(str)
```

```
    binary = reshape(dec2bin(str, 8).', 1, []);
```

```
end
```

% Function to embed the message using LSB substitution / LSB प्रतिस्थापन का उपयोग करके संदेश दबाने के लिए फ़ंक्शन

```
function stegoImage = embedMessage(coverImage, message)
```

```
    stegoImage = coverImage;
```

```
    [rows, cols, ~] = size(coverImage);
```

```
    messageIndex = 1;
```

```
    for row = 1:rows
```

```
        for col = 1:cols
```

```
            pixel = stegoImage(row, col, :);
```

```
            for channel = 1:3 % For RGB channels
```

```
                if messageIndex <= length(message)
```

```
                    pixel(channel) = bitset(pixel(channel), 1, message(messageIndex) == '1');
```

```
                    messageIndex = messageIndex + 1;
```

```
                end
```

```
            end
```

```
            stegoImage(row, col, :) = pixel;
```

```
        end
```



```
end  
end  
  
% Function to extract the message using LSB extraction / LSB निष्कर्षण का उपयोग करके संदेश का निष्कर्षण  
करने के लिए फ़ंक्शन  
function extractedMessage = extractMessage(stegoImage)  
    extractedMessage = "";  
    [rows, cols, ~] = size(stegoImage);  
  
    for row = 1:rows  
        for col = 1:cols  
            pixel = stegoImage(row, col, :);  
            for channel = 1:3 % For RGB channels  
                extractedMessage = [extractedMessage, num2str(bitget(pixel(channel), 1))];  
            end  
        end  
    end  
end  
  
% Convert binary message to characters / बाइनरी संदेश को वर्णों में रूपांतरित करें  
extractedMessage = char(bin2dec(reshape(extractedMessage, 8, [])).');  
end
```



हफ़ ट्रांसफ़ॉर्म का उपयोग करके ऑब्जेक्ट की सीमा का पता लगाने के लिए एक प्रोग्राम लिखें।

Write a program to detect the boundary of the object using Hough Transform.

% Load and preprocess the image / छवि लोड करें और पूर्व-प्रसंस्करण करें

```
originalImage = imread('your_image_path.jpg');
```

```
grayImage = rgb2gray(originalImage);
```

```
edgeImage = edge(grayImage, 'canny');
```

% Define parameters for Hough Transform / Hough परिवर्तन के लिए पैरामीटरों की परिभाषा करें

```
minRadius = 20;
```

```
maxRadius = 100;
```

% Initialize accumulator array / एक्क्यूमुलेटर मास्क इनिशियलाइज़ करें

```
accumulator = zeros(size(edgeImage));
```

% Perform Hough Transform / हफ़ परिवर्तन कार्यान्वित करें

```
for row = 1:size(edgeImage, 1)
```

```
    for col = 1:size(edgeImage, 2)
```

```
        if edgeImage(row, col) == 1
```

```
            for radius = minRadius:maxRadius
```

```
                for theta = 0:0.01:2*pi
```

```
                    x = col - radius * cos(theta);
```

```
                    y = row - radius * sin(theta);
```

```
                    if x >= 1 && x <= size(edgeImage, 2) && y >= 1 && y <= size(edgeImage, 1)
```

```
                        accumulator(round(y), round(x)) = accumulator(round(y), round(x)) + 1;
```

```
                    end
```



```
end
end
end
end
end

% Find peaks in the accumulator array / एक्क्यूमुलेटर मास्क में चोटियाँ ढूँढें
threshold = 0.5 * max(accumulator(:));
[centersRow, centersCol] = find(accumulator > threshold);

% Draw detected circles on the original image / मूल छवि पर पहचानी गई वृत्तों को ड्रा करें
for i = 1:length(centersRow)
    center = [centersCol(i), centersRow(i)];
    radius = minRadius + (maxRadius - minRadius) * rand(); % Random radius for visualization
    originalImage = insertShape(originalImage, 'Circle', [center, radius], 'LineWidth', 2, 'Color', 'red');
end

% Display the result / परिणाम दिखाएँ
imshow(originalImage);

title('Detected Circles using Hough Transform / Hough परिवर्तन का उपयोग करके पहचाने गए वृत्त');
```

लैब असाइनमेंट १२

Lab Assignment 12

छवि के ज्यामितीय रूपांतरण के लिए प्रोग्राम लिखें और निष्पादित करें।

write and execute program for geometric transformation of image



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

% Load the original image / मूल छवि लोड करें

imagePath = 'आपकी_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your image file path / अपनी छवि फ़ाइल के पथ से बदलें

originalImage = imread(imagePath);

% a. Translation / स्थानांतरण

tx = 50; % Translation in x-direction / x-दिशा में स्थानांतरण

ty = 30; % Translation in y-direction / y-दिशा में स्थानांतरण

translatedImage = imtranslate(originalImage, [tx, ty]); % Perform translation / स्थानांतरण कार्यान्वित करें

imshow(translatedImage);

title('Translated Image / स्थानांतरित छवि');

% b. Scaling / पैमाना बदलें

scaleFactor = 1.5; % Scaling factor / पैमाना गुणक

scaledImage = imresize(originalImage, scaleFactor); % Perform scaling / पैमाना बदलें

imshow(scaledImage);

title('Scaled Image / पैमाने की छवि');

% c. Rotation / घुमाना

angle = 30; % Rotation angle in degrees / घुमाने का कोण (डिग्री में)

rotatedImage = imrotate(originalImage, angle); % Perform rotation / घुमाना कार्यान्वित करें

imshow(rotatedImage);



```
title('Rotated Image / घुमाई गई छवि');
```

% d. Shrinking / छोटा करना

```
shrinkFactor = 0.7; % Shrinking factor / छोटा करने का गुणक
```

```
shrunkImage = imresize(originalImage, shrinkFactor); % Perform shrinking / छोटा करना
```

```
imshow(shrunkImage);
```

```
title('Shrunk Image / छोटी छवि');
```

% e. Zooming: zero order, first order, and interpolation method

% जूमिंग: जीरो आर्डर, पहले आर्डर और अंतर्पोलेशन विधि

```
zoomedZeroOrder = imresize(originalImage, 1.5, 'nearest'); % Zero order / जीरो आर्डर
```

```
zoomedFirstOrder = imresize(originalImage, 1.5, 'bilinear'); % First order / पहले आर्डर
```

```
zoomedInterpolation = imresize(originalImage, 1.5, 'bicubic'); % Interpolation / अंतर्पोलेशन
```

```
figure;
```

```
subplot(1, 3, 1), imshow(zoomedZeroOrder), title('Zoomed (Zero Order) / जूमिंग (जीरो आर्डर)');
```

```
subplot(1, 3, 2), imshow(zoomedFirstOrder), title('Zoomed (First Order) / जूमिंग (पहले आर्डर)');
```

```
subplot(1, 3, 3), imshow(zoomedInterpolation), title('Zoomed (Interpolation) / जूमिंग (अंतर्पोलेशन)');
```

लैब असाइनमेंट 13

Lab Assignment 13

विभिन्न छवि नॉइसमॉडल को समझना और छवि को पुनर्स्थापित करने के लिए प्रोग्राम लिखें।

Write a program to understand and detect the noise present in an image and remove them.



% Load the noisy image / नॉइसयुक्त छवि लोड करें

imagePath = 'आपकी_शोरयुक्त_छवि_फ़ाइल_का_पथ.jpg'; % Replace with your noisy image file path / अपनी शोरयुक्त छवि फ़ाइल के पथ से बदलें

noisyImage = imread(imagePath);

% a. Remove Salt and Pepper Noise / साल्ट एवं पेप्पर के नॉइस को हटाएं

saltAndPepperRemoved = medfilt2(noisyImage, [3, 3]); % Apply median filter

imshow(saltAndPepperRemoved);

title('Salt and Pepper Noise Removed / साल्ट एवं पेप्पर के नॉइस हटाई गई');

% b. Minimize Gaussian Noise / गॉसियन नॉइस कम करें

gaussianNoiseMinimized = imgaussfilt(noisyImage, 2); % Apply Gaussian filter

imshow(gaussianNoiseMinimized);

title('Gaussian Noise Minimized / गॉसियन नॉइस कम हुई');

% c. Periodic Noise

% Define parameters for creating periodic noise / आवधिक नॉइस बनाने के लिए पैरामीटर निर्धारित करें

periodicNoiseFrequency = 0.05; % Frequency of the periodic noise / आवधिक नॉइस का आवृत्ति

[M, N] = size(noisyImage);

[X, Y] = meshgrid(1:N, 1:M);

periodicNoise = sin(2 * pi * periodicNoiseFrequency * X); % Generate the periodic noise / आवधिक नॉइस उत्पन्न करें

periodicNoisyImage = noisyImage + uint8(50 * periodicNoise); % Add periodic noise to the image / आवधिक नॉइस को छवि में जोड़ें



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)

```
periodicNoiseRemoved = wiener2(periodicNoisyImage, [5, 5]); % Apply Wiener filter for periodic noise removal  
imshow(periodicNoiseRemoved);  
  
title('Periodic Noise Removed / आवधिक नॉइस हटाई गई');
```

विषय समन्वयक (Subject Coordinators)

डॉ. ज्योति भारती (Dr. Jyoti Bharti)

डॉ. रमेश कुमार ठाकुर (Dr. Ramesh Kumar Thakur)

डॉ. नमिता तिवारी (Dr. Namita Tiwari)

शिक्षण सहायक (Teacher Assistants)

PhD 3 - Richa Goenka, PhD 11 - Sameeksha Prasad (Sec- CSE-1)

Guest Faculty – SBS, BP (Sec- CSE-1, CSE - 3)

Guest Faculty – SBS, MUG (Sec- CSE-2)



मौलाना आज़ाद राष्ट्रीय प्रौद्योगिकी संस्थान, भोपाल – 462003
MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL – 462003

कंप्यूटर विज्ञान एवं अभियांत्रिकी विभाग
(Department of Computer Science and Engineering)