A logo for a university

AI-generated content may be incorrect.

**Lab Report 7 & 8**

**Digital Image Processing**

**CSE438**

**Section:** 03

**Semester:** Spring-2025

**Submitted To:**

**Md. Asif Khan Rifat**

**Lecturer**

Department of Computer Science

and Engineering

**Submitted By:**

                                                                         Suddip Paul Arnab

**2022-1-60-356**

**Date of submission:** 8 May 2025

**LAB 07 AND 08**

1. **Detect** the tumor from the images using the segmentation approaches listed below:

(Outline the segmented object to highlight the tumor. You can crop the image for

accurate segmentation.)

i) Similarity approaches:

a) Local/Regional Thresholding

b) Global Thresholding

c) Variable Thresholding

d) Dynamic/Adaptive Thresholding

ii) Discontinuity approaches: Edge Detection (Sobel, Canny, Prewitt)

img1 = imread('Picture1.png');

img2 = imread('Picture2.png');

gray\_img1 = im2gray(img1);

gray\_img2 = im2gray(img2);

figure;

subplot(3,4,1); imshow(img1); title('Original Image 1');

subplot(3,4,2); imshow(img2); title('Original Image 2');

bw\_local1 = gray\_img1 > 120;

bw\_local2 = gray\_img2 > 120;

subplot(3,4,3); imshow(bw\_local1); title('Local Thresholding 1');

subplot(3,4,4); imshow(bw\_local2); title('Local Thresholding 2');

level1 = graythresh(gray\_img1);

level2 = graythresh(gray\_img2);

bw\_global1 = imbinarize(gray\_img1, level1);

bw\_global2 = imbinarize(gray\_img2, level2);

subplot(3,4,5); imshow(bw\_global1); title('Global Thresholding 1');

subplot(3,4,6); imshow(bw\_global2); title('Global Thresholding 2');

adaptive\_thresh1 = adaptthresh(gray\_img1, 0.5);

adaptive\_thresh2 = adaptthresh(gray\_img2, 0.5);

bw\_variable1 = imbinarize(gray\_img1, adaptive\_thresh1);

bw\_variable2 = imbinarize(gray\_img2, adaptive\_thresh2);

subplot(3,4,7); imshow(bw\_variable1); title('Variable Thresholding 1');

subplot(3,4,8); imshow(bw\_variable2); title('Variable Thresholding 2');

bw\_dynamic1 = imbinarize(gray\_img1, adaptthresh(gray\_img1));

bw\_dynamic2 = imbinarize(gray\_img2, adaptthresh(gray\_img2));

subplot(3,4,9); imshow(bw\_dynamic1); title('Dynamic Thresholding 1');

subplot(3,4,10); imshow(bw\_dynamic2); title('Dynamic Thresholding 2');

edges\_sobel1 = edge(gray\_img1, 'sobel');

edges\_sobel2 = edge(gray\_img2, 'sobel');

subplot(3,4,11); imshow(edges\_sobel1); title('Sobel Edge Detection 1');

subplot(3,4,12); imshow(edges\_sobel2); title('Sobel Edge Detection 2');

figure;

edges\_canny1 = edge(gray\_img1, 'canny');

edges\_canny2 = edge(gray\_img2, 'canny');

subplot(2,3,1); imshow(edges\_canny1); title('Canny Edge Detection 1');

subplot(2,3,2); imshow(edges\_canny2); title('Canny Edge Detection 2');

edges\_prewitt1 = edge(gray\_img1, 'prewitt');

edges\_prewitt2 = edge(gray\_img2, 'prewitt');

subplot(2,3,3); imshow(edges\_prewitt1); title('Prewitt Edge Detection 1');

subplot(2,3,4); imshow(edges\_prewitt2); title('Prewitt Edge Detection 2');

tumor\_outline1 = bwperim(bw\_dynamic1);

tumor\_outline2 = bwperim(bw\_dynamic2);

overlay1 = imoverlay(gray\_img1, tumor\_outline1, [1 0 0]); % Red outline

overlay2 = imoverlay(gray\_img2, tumor\_outline2, [1 0 0]); % Red outline

subplot(2,3,5); imshow(overlay1); title('Tumor Outline 1');

subplot(2,3,6); imshow(overlay2); title('Tumor Outline 2');

disp('Tumor detection and segmentation complete!');

A group of images of a brain

AI-generated content may be incorrect.

A group of images of a brain

AI-generated content may be incorrect.

2. **Show** in a table how the Similarity and Discontinuity techniques compare.

segmentation, based on their effectiveness, strengths, weaknesses, and suitability for isolating and outlining the tumor.

| **Technique** | **Type** | **Description** | **Strengths** | **Weaknesses** | **Observatiom** |
| --- | --- | --- | --- | --- | --- |
| **Local Thresholding** | Similarity | Applies a fixed threshold (e.g., 120) to segment the image based on intensity. | Simple to implement; effective if tumor has consistent intensity. | Sensitive to lighting variations; may miss tumors with varying intensity. | Moderate; works if tumor intensity is distinct and uniform. |
| **Global Thresholding** | Similarity | Uses Otsu’s method to find a single optimal threshold for the entire image. | Automatic threshold selection; robust for images with clear bimodal histograms. | Fails with non-uniform lighting or complex backgrounds; may include non-tumor areas. | Good; effective for images with clear tumor-background contrast. |
| **Variable Thresholding** | Similarity | Adaptive thresholding with a specified sensitivity (e.g., 0.5) to adjust locally. | Handles local intensity variations; more flexible than global thresholding. | Requires sensitivity tuning; may produce noisy masks in complex images. | Very good; suitable for tumors with local intensity variations. |
| **Dynamic/Adaptive Thresholding** | Similarity | Adaptive thresholding with default sensitivity, adjusting based on local neighborhoods. | Highly adaptive to local variations; produces clean masks with proper tuning. | Computationally intensive; sensitive to parameter settings (e.g., window size). | Excellent; best for complex images with varying tumor intensity. |
| **Sobel Edge Detection** | Discontinuity | Detects edges using Sobel operators to highlight intensity gradients. | Simple; effective for detecting strong tumor boundaries. | Sensitive to noise; produces incomplete boundaries; not a full segmentation. | Poor; outlines edges but doesn’t segment the tumor region. |
| **Canny Edge Detection** | Discontinuity | Advanced edge detection with noise suppression and edge linking. | Robust to noise; produces thin, connected edges; good for clear boundaries. | Requires post-processing to form closed regions; not a full segmentation. | Moderate; good for outlining but incomplete for tumor segmentation. |
| **Prewitt Edge Detection** | Discontinuity | Detects edges using Prewitt operators, similar to Sobel but with different kernels. | Simple; similar to Sobel but slightly less sensitive to noise. | Like Sobel, produces incomplete boundaries; sensitive to noise. | Poor; similar limitations to Sobel, unsuitable for full segmentation. |

3. Generate a binary mask of the tumor from Figure 1 using any segmentation method of your choice, then apply:

i. Morphological Dilation

ii. Morphological Erosion

By using appropriate structuring element on the mask.

img = imread('Picture3.png');

gray\_img = im2gray(img);

binary\_mask = imbinarize(gray\_img, adaptthresh(gray\_img));

figure;

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

subplot(1,3,2); imshow(binary\_mask); title('Binary Mask');

se = strel('disk', 5);

dilated\_mask = imdilate(binary\_mask, se);

subplot(1,3,3); imshow(dilated\_mask); title('Dilated Mask');

figure;

eroded\_mask = imerode(binary\_mask, se);

subplot(1,2,1); imshow(dilated\_mask); title('After Dilation');

subplot(1,2,2); imshow(eroded\_mask); title('After Erosion');

disp('Binary mask and morphological processing completed!');

A close-up of a brain

AI-generated content may be incorrect.

A close-up of a brain

AI-generated content may be incorrect.

4. **Apply** the Hough transform to Figure 2 and draw the detected lines on the original image.

img = imread('Picture4.jpg');

gray\_img = rgb2gray(img);

edges = edge(gray\_img, 'Canny');

[H, theta, rho] = hough(edges);

peaks = houghpeaks(H, 10, 'Threshold', ceil(0.3 \* max(H(:))));

lines = houghlines(edges, theta, rho, peaks, 'FillGap', 20, 'MinLength', 30);

figure;

subplot(1,3,1);

imshow(img);

title('Original X-ray');

subplot(1,3,2);

imshow(edges);

title('Edge Detection');

subplot(1,3,3);

imshow(img);

title('Detected Lines');

hold on;

for k = 1:length(lines)

xy = [lines(k).point1; lines(k).point2];

line(xy(:,1), xy(:,2), 'Color','green','LineWidth',2);

end

hold off;

A hand with dots on it

AI-generated content may be incorrect.