EAST

EB East West University

Course title: Digital Image Processing  
Course code: CSE438

Spring 2023

Section: 1

LAB REPORT 6

Submitted To:

Dr. Ahmed Wasif Reza

Professor, Department of CSE

Additional Director, Institutional Quality Assurance Cell (IQAC)  
East West University, Dhaka

Submitted By:

K. M. Safin Kamal  
ID: 2020-1-60-235

Department of Computer Science and Engineering  
East West University, Dhaka

Date of Submission: 14/ 05 / 2023

Q1. segment the tumor from figure 1 by using:

1. Region growing approach
2. Region Splitting and Merging approach

function J=regiongrowing(I,x,y,reg\_maxdist)

if(exist('reg\_maxdist','var')==0), reg\_maxdist=0.2; end

if(exist('y','var')==0), figure, imshow(I,[]); [y,x]=getpts; y=round(y(1)); x=round(x(1)); end

J = zeros(size(I));

Isizes = size(I);

reg\_mean = I(x,y);

reg\_size = 1;

neg\_free = 10000; neg\_pos=0;

neg\_list = zeros(neg\_free,3);

pixdist=0;

neigb=[-1 0; 1 0; 0 -1;0 1];

diff = 01;

while(pixdist<reg\_maxdist && reg\_size<numel(I) && diff ~=0) num1 = sum(sum(reg\_size));

% Add new neighbors pixels for j=1:4,

% Calculate the neighbour coordinate xn = x +neigb(j,1); yn = y +neigb(j,2);

% Check if neighbour is inside or outside the image ins=(xn>=1)&&(yn>=1)&&(xn<=Isizes(1))&&(yn<=Isizes(2));

% Add neighbor if inside and not already part of the segmented area if(ins&&(J(xn,yn)==0))

neg\_pos = neg\_pos+1;

neg\_list(neg\_pos,:) = [xn yn I(xn,yn)]; J(xn,yn)=1; end end

if(neg\_pos+10>neg\_free), neg\_free=neg\_free+10000;

neg\_list((neg\_pos+1):neg\_free,:)=0; end

dist = abs(neg\_list(1:neg\_pos,3)-reg\_mean);

[pixdist, index] = min(dist);

J(x,y)=2; reg\_size=reg\_size+1;

reg\_mean= (reg\_mean\*reg\_size + neg\_list(index,3))/(reg\_size+1);

x = neg\_list(index,1); y = neg\_list(index,2);

neg\_list(index,:)=neg\_list(neg\_pos,:); neg\_pos=neg\_pos-1;

num2 = sum(sum(reg\_size));

diff = num2-num1;

end

J=J>1;

I = rgb2gray(im2double(imread("D:\UNIVERSITY\Semester 10 (cse 400A,438,475,487)\CSE438\Lab\LAB6\Tumor (1).png")));

J=imadjust(I,[],[],0.5);

J=imgaussfilt(J);

J= regiongrowing(J,0.2);

imshowpair(I,J,'montage')

figure;

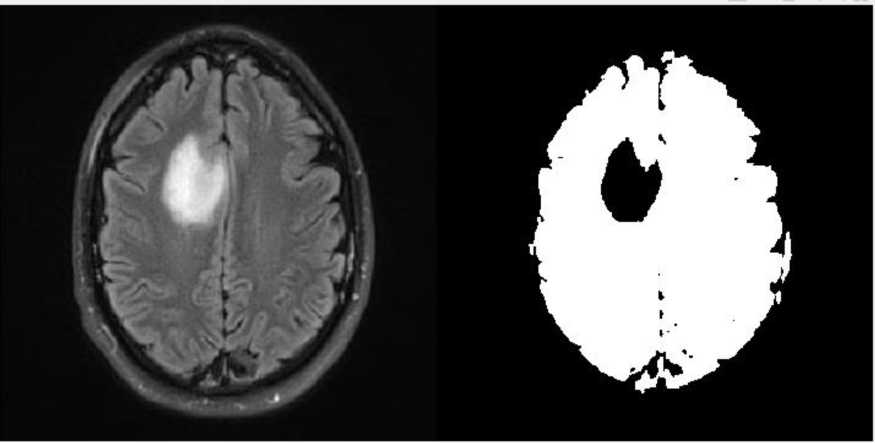
subplot(1, 3, 1); imshow(img); title('Original Image');

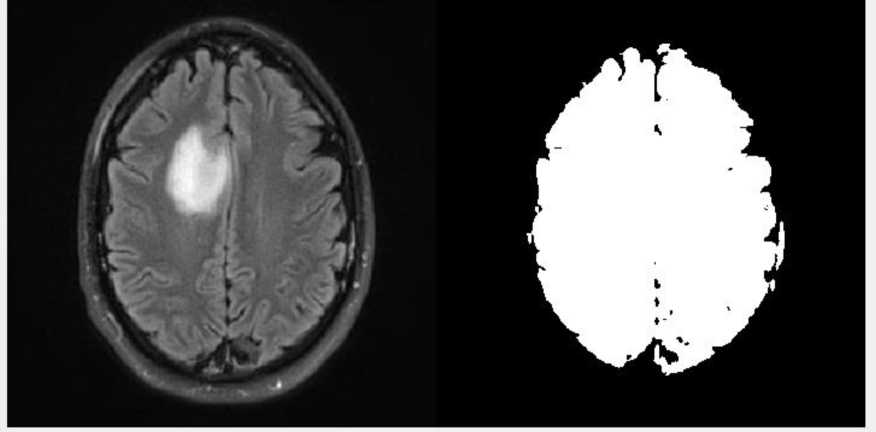
subplot(1, 3, 2); imshow(tumor\_mask\_rg); title('Region Growing

Segmentation');

subplot(1, 3, 3); imshow(tumor\_mask\_rsm); title('Region Splitting and Merging

Segmentation');





Q2. examine the differences between the methods from question 1 to highlight their differences in quality and accessibility.

Based on the results obtained from the segmentation of the provided image in Question 1, the region growth method appears to be effective in highlighting and isolating the tumor in black, and it also generates a corresponding mask. On the other hand, the split-and-merge technique was not able to accurately segment the tumor from the brain. Hence, in this scenario, the region growth method is deemed to be a more successful approach for tumor segmentation.

Q3. segment the tumor from figure 1 by using marker controlled watershed segmentation

img = imread("D:\UNIVERSITY\Semester 10 (cse 400A,438,475,487)\CSE438\Lab\LAB6\Tumor (1).png");

% convert to grayscale gray\_img = rgb2gray(img);

% threshold the image to get a binary mask thresh = graythresh(gray\_img);

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

% perform morphological operations to enhance the mask se = strel('disk', 5);

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

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

% compute the distance transform of the mask distance\_transform = -bwdist(~dilated\_mask);

% apply the watershed algorithm

watershed\_lines = watershed(distance\_transform);

% identify the markers within the watershed lines markers = imregionalmin(distance\_transform);

% apply the markers to the watershed lines

marked\_watershed = watershed(imimposemin(distance\_transform, markers));

% overlay the segmented tumor on the original image

tumor\_region = gray\_img;

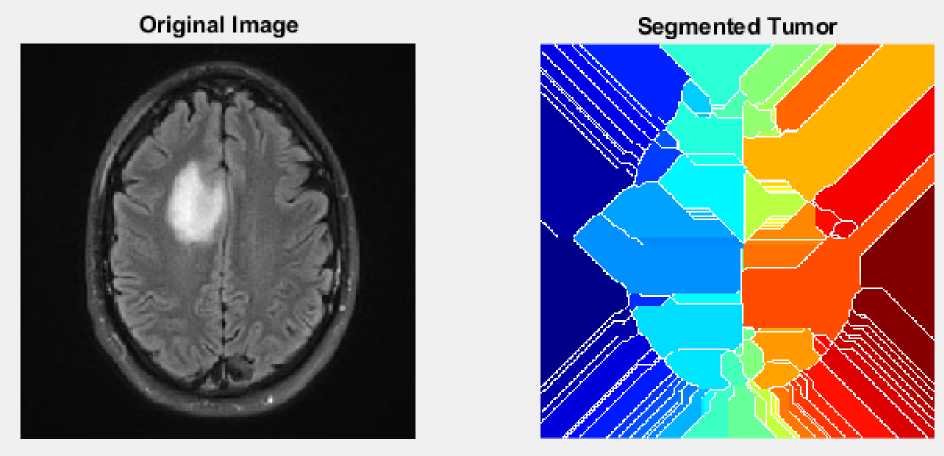
tumor\_region(marked\_watershed == 0) = 0;

tumor\_segmented = label2rgb(marked\_watershed, 'jet', 'w');

% display the results

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

subplot(1,2,2), imshow(tumor\_segmented), title('Segmented Tumor');



Q4. segment the tumor from figure 1 by using quadtree segmentation

% Load the image

img = imread("D:\UNIVERSITY\Semester 10 (cse

400A,438,475,487)\CSE438\Lab\LAB6\Tumor (1).png");

I=rgb2gray(img);

Ifilt = medfilt2(I,[8 8]);

1. = qtdecomp\_var(I,10);

[Sfilt] = qtdecomp\_var(Ifilt);

blocks = repmat(uint8(0),size(S));

for dim = [256 128 64 32 16 8 4 2 1]

numblocks = length(find(S==dim)); if (numblocks > 0)

values = repmat(uint8(1),[dim dim numblocks]);

values(2:dim,2:dim,:) = 0;

blocks = qtsetblk(blocks,S,dim,values); end

end

blocks(end,1:end) = 1;

blocks(1:end,end) = 1;

blocks\_filt = repmat(uint8(0),size(Sfilt));

for dim = [128 64 32 16 8 4 2 1]

numblocks = length(find(Sfilt==dim));

if (numblocks > 0)

values = repmat(uint8(1),[dim dim numblocks]);

values(2:dim,2:dim,:) = 0;

blocks\_filt = qtsetblk(blocks\_filt,Sfilt,dim,values); end

end

blocks\_filt(end,1:end) = 1;

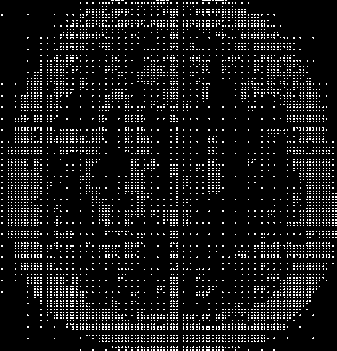
blocks\_filt(1:end,end) = 1;

figure;subplot(221);imshow(I); title('Input image');

subplot(222);imshow(Ifilt); title('Filtered image');

subplot(223);imshow(blocks,[]);title('Quad tree input image with var weight');

subplot(224);imshow(blocks\_filt,[]);title('Quad tree filtered image without var weight');



Q5. generate a binary mask of the tumor from figure 1 using any segmentation method of your choice, then apply:

1. Morphological Dilation
2. Morphological Erosion

By using appropriate structuring element on the mask.

% Load the image

img = imread("D:\UNIVERSITY\Semester 10 (cse

400A,438,475,487)\CSE438\Lab\LAB6\Tumor (1).png");

% Convert to grayscale gray\_img = rgb2gray(img);

% Threshold using Otsu's method level = graythresh(gray\_img);

binary\_img = imbinarize(gray\_img, level);

% Apply morphological dilation and erosion

se = strel('disk', 10); % create a disk-shaped structuring element of radius

10 pixels

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

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

% Display the results

figure;

subplot(2,2,1);

imshow(img);

title('Original Image');

subplot(2,2,2);

imshow(binary\_img);

title('Binary Mask');

subplot(2,2,3);

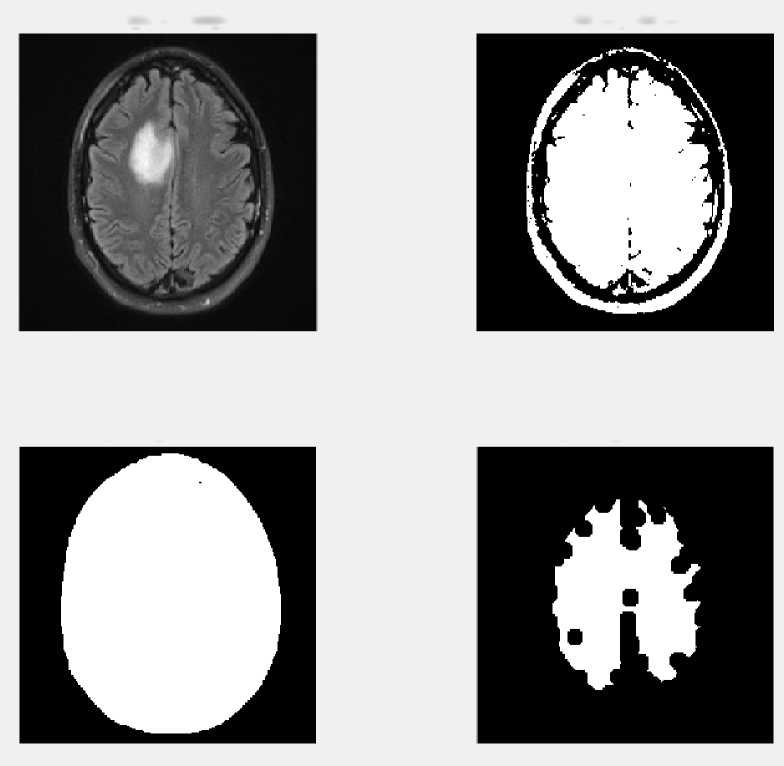
imshow(dilated\_img);

title('Morphological Dilation');

subplot(2,2,4);

imshow(eroded\_img);

title('Morphological Erosion');



Original Image

Morphological Dilation

Binary Mask

Morphological Erosion

Q6. apply hough transform to figure 2 and draw the detected lines on the original image.

% read the image

img = imread("D:\UNIVERSITY\Semester 10 (cse

400A,438,475,487)\CSE438\Lab\LAB6\X-Ray Image.jpg");

grayImg = rgb2gray(img);

edgeImg = edge(grayImg, 'Canny');

[H,theta,rho] = hough(edgeImg,'Theta',-90:0.5:89.5);

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

lines = houghlines(edgeImg,theta,rho,peaks,'FillGap',10,'MinLength',20);

imshow(img);

hold on;

for k = 1:length(lines)

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

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

end

