**east west university**

**Lab Report - 08**

**Department:** **Computer Science and Engineering**

**Course Title:** Digital Image Processing

**Course Code:** CSE438

**Section No:** 02

**Submitted To**:

Dr. Engr. Ahmed Wasif Reza

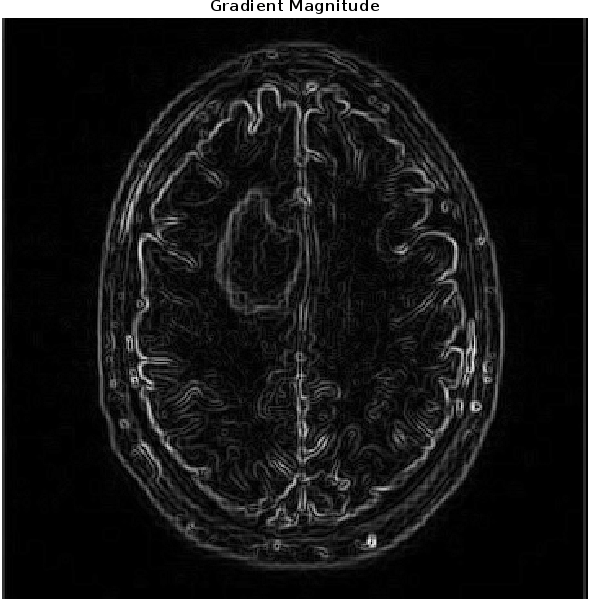
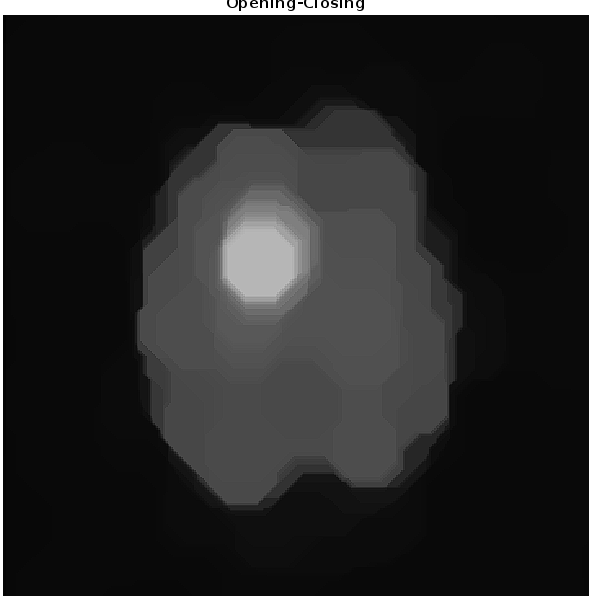
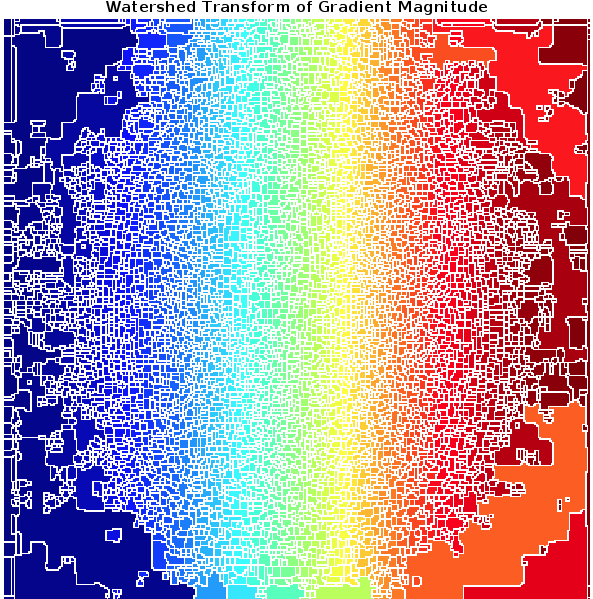
Associate Professor, Department of Computer Science and Engineering

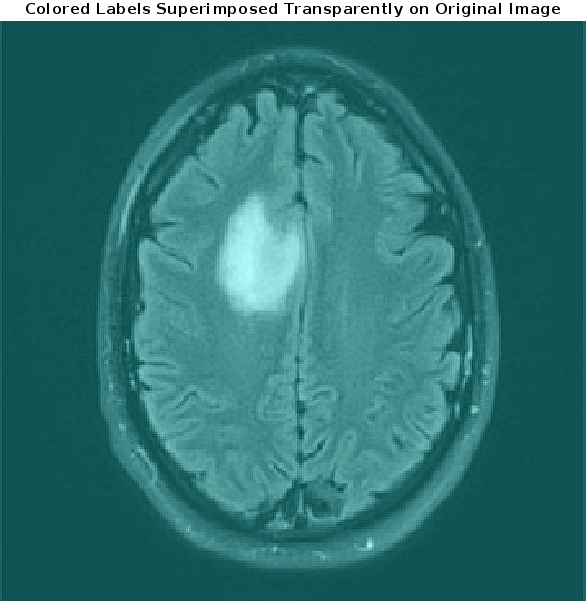
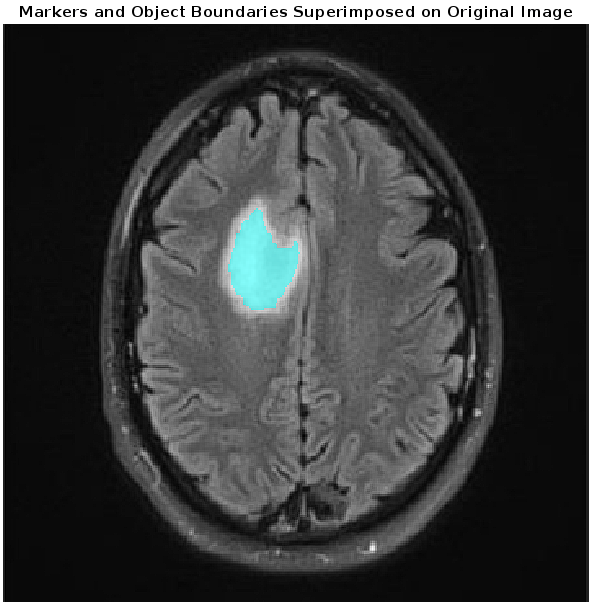
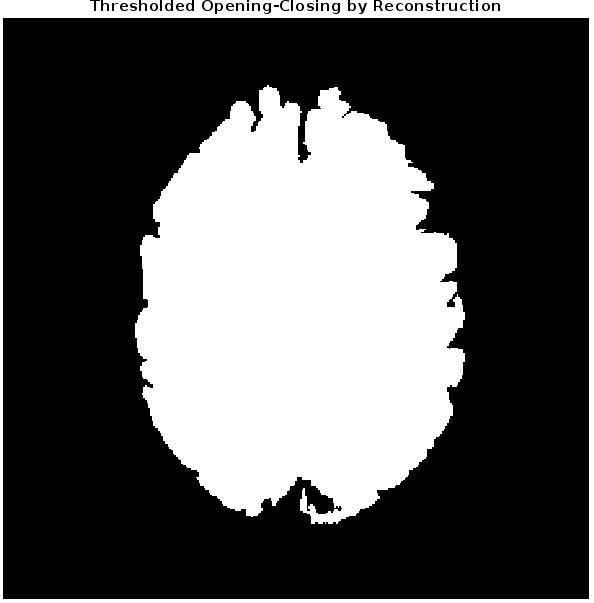
**Submitted By**:

Name: S M Arafat Rahman

ID: **2019-2-60-094**

2. Segment the tumor from Figure 1 by using Marker Controlled Watershed segmentation.





Code:

rgb = imread("Tumor.png");

I = rgb2gray(rgb);

figure, imshow(I)

gmag = imgradient(I);

figure, imshow(gmag,[])

title('Gradient Magnitude')

L = watershed(gmag);

Lrgb = label2rgb(L);

figure, imshow(Lrgb)

title('Watershed Transform of Gradient Magnitude')

se = strel('disk',20);

Io = imopen(I,se);

figure, imshow(Io)

title('Opening')

Ie = imerode(I,se);

Iobr = imreconstruct(Ie,I);

figure, imshow(Iobr)

title('Opening-by-Reconstruction')

Ioc = imclose(Io,se);

figure, imshow(Ioc)

title('Opening-Closing')

Iobrd = imdilate(Iobr,se);

Iobrcbr = imreconstruct(imcomplement(Iobrd),imcomplement(Iobr));

Iobrcbr = imcomplement(Iobrcbr);

figure, imshow(Iobrcbr)

title('Opening-Closing by Reconstruction')

fgm = imregionalmax(Iobrcbr);

figure, imshow(fgm)

title('Regional Maxima of Opening-Closing by Reconstruction')

I2 = labeloverlay(I,fgm);

figure, imshow(I2)

title('Regional Maxima Superimposed on Original Image')

se2 = strel(ones(5,5));

fgm2 = imclose(fgm,se2);

fgm3 = imerode(fgm2,se2);

fgm4 = bwareaopen(fgm3,20);

I3 = labeloverlay(I,fgm4);

figure, imshow(I3)

title('Modified Regional Maxima Superimposed on Original Image')

bw = imbinarize(Iobrcbr);

figure, imshow(bw)

title('Thresholded Opening-Closing by Reconstruction')

D = bwdist(bw);

DL = watershed(D);

bgm = DL == 0;

figure, imshow(bgm)

title('Watershed Ridge Lines')

gmag2 = imimposemin(gmag, bgm | fgm4);

L = watershed(gmag2);

labels = imdilate(L==0,ones(3,3)) + 2\*bgm + 3\*fgm4;

I4 = labeloverlay(I,labels);

figure, imshow(I4)

title('Markers and Object Boundaries Superimposed on Original Image')

Lrgb = label2rgb(L,'jet','w','shuffle');

figure, imshow(Lrgb)

title('Colored Watershed Label Matrix')

figure, imshow(I)

hold on

himage = imshow(Lrgb);

himage.AlphaData = 0.3;

title('Colored Labels Superimposed Transparently on Original Image')

3. Segment the tumor from Figure 1 by using Quadtree Segmentation.

I=imread('Tumor.png');

Ig = rgb2gray(I);size(Ig)

S = qtdecomp(Ig, .27);

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

for dim = [512 256 128 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;

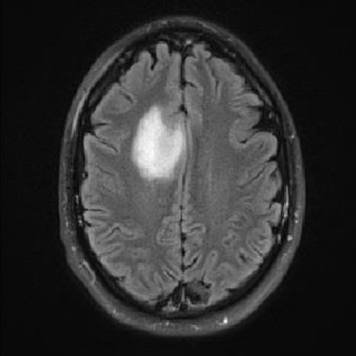
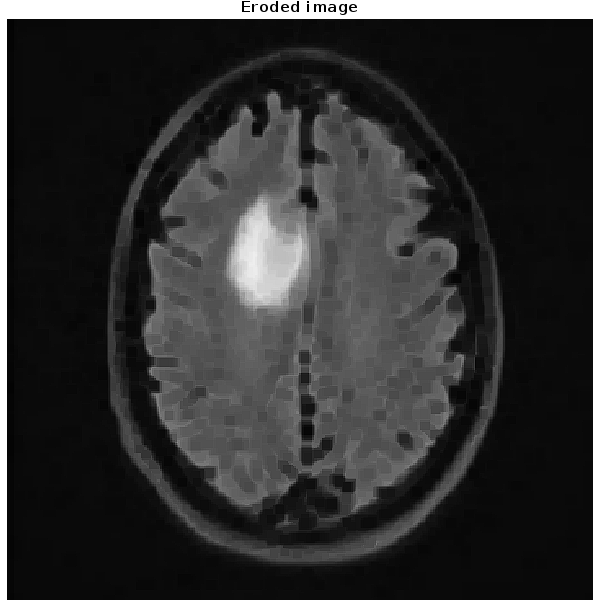
blocks(1:end,end) = 1;

imshow(I)

4. 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 elements on the mask.



1. Segment the tumor from Figure 1 by using:

i. Region-growing approach

ii. Region Splitting and Merging approach

Code:

I = im2double(imread('Tumor.png'));

figure, imshow(I)

Isizes = size(I);

threshI = multithresh(I, 3);

[m, n]=ginput(1);

c = impixel(I, m, n);

currPix = c(1);

surr = [-1 0; 1 0; 0 -1; 0 1];

mem = zeros(Isizes(1)\*Isizes(2), 3); and pixel value

mem(1, :) = [m, n, currPix];

regSize = 1;

J = zeros(Isizes(1), Isizes(2));

init = 1;

posInList = 1;

k=1;

while(k==1)

for l=init:posInList

for j=1:4

m1 = m + surr(j,1);

n1 = n + surr(j,2);

check=(m1>=1)&&(n1>=1)&&(m1<=Isizes(1))&&(n1<=Isizes(2));

current = impixel(I, m1, n1);

currPix = current(1);

if(check && currPix<=threshI(2) && (J(m1, n1)==0))

posInList = posInList+1;

mem(posInList, :) = [m1, n1, currPix];

J(m1, n1) = 1;

end

end

end

if(posInList == init)

k = 0;

else

init = init+1;

m = mem(init, 1, :);

n = mem(init, 2, :);

k = 1;

end

end

figure, imshow(J);

