**BRAIN TUMOR DETECTION USING IMAGE PROCESSING**

**ABSTRACT**

Brain tumor detection is the most difficult and tedious task in medical imaging. Magnetic resonance imaging (MRI) can be a medical procedure that is generally used by the medical specialist to show the internal structure of the structure without surgery. Magnetic resonance imaging provides a lot of information about sensitive human tissue, which helps infer brain tumors. Accurate segmentation of the

MRI image is critical for concluding a brain tumor with a laptop-compatible clinical device. This report focuses on the best and most correct approach to detecting neoplasms by brain MRI and, if it confirms the presence of a tumor, focuses on evaluating its stage. We conducted an experiment. Showed that our proposed methodology offers greater precision than the various existing strategies for tumor detection.

**INTRODUCTION**

This project proposes to segment a tumor from a magnetic resonance image and identify the tumor. For this purpose, segmentation and grouping techniques were implemented. Each MRI image is passed through an image chain where the image is pre-processed to remove noise and improve image contrast. The application of each of the segmentation techniques allows us to determine the most appropriate method of segmenting the tumor for each of the images. In this project, the most important challenge was to locate and extract the correct tumor region from the image. Due to various lighting issues, there were unnecessary white areas in the image that could be incorrectly segmented as a tumor. In addition, unwanted noise and reduced contrast reveal several areas of the image that are incorrectly labeled as a tumor. Another challenge was the degraded quality of the MRI image due to various issues that would have arisen during the acquisition phase.

**PROPOSED METHOD**

In this project we have described our goal as brain tumor detection, ie the presence of the tumor on the provided MRI. Here we will analyze the MRI images that conclude the stage of the tumor. In general, the diagram for our process. The input images go through several stages that can be summarized as shown in Figure 1.

MRI of Brain Images



Pre-Processing



Feature Extraction



Segmentation Technique



Image Analysis

Diagram

Description automatically generated**MRI of Brain images**

**Figure 1 Brain tumor detection steps**

This is the first step of our proposed project .In this the data has been provided that is the magnetic resonance images(MRI) that are been collected in their original format’s that are (.ima, .dcm). Mostly the mri images are of .dcm (DICOM[13]) Digital imaging and communications in medicine. We have used file operations fopen(), fclose() available in matlab to read MRI images. Here the gray scale MRI images are being provided as input to the system.

**Pre-Processing**

Pre-processing phase of our project mainly involves those operations that are ordinarily essential before the goal analysis and extraction of the required data and ordinarily geometric corrections of the initial image. These enhancements embrace correcting the information for irregularities and unwanted region noise, removal of non-brain element image and converting the data so that they are correctly reflected in the original image. The first step of pre- processing is the conversion of the given input MRI image into a suitable form on which further work can be performed.

This conversion of DICOM image to .jpeg is done by using function dicom2image()[7]. Major issues related to the pre- processing stage are as follows:-

* Noise,
* Blur Low Contrast,
* The partial-volume effect.

This pre-processing stage is used for reducing image noise, highlighting important portions, or displaying obvious portions of digital images.

**De-noising method**

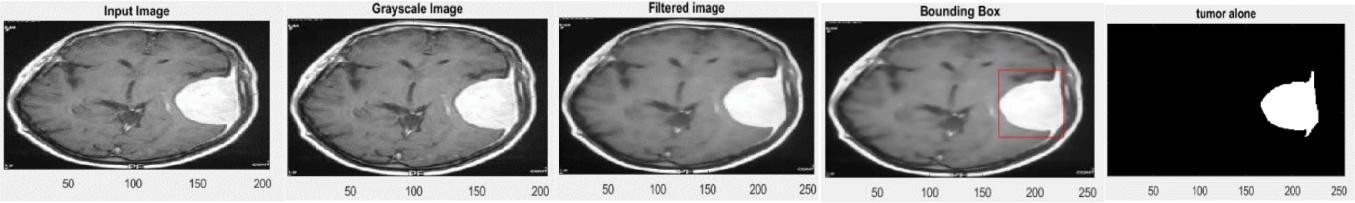
Despite the presence of considerable variety of state-of-the-art ways of de-noising however correct removal of noise from magnetic resonance imaging image is a challenge. Here, Filter based method has been used as a de-noising. We have used the Filter transform functions that is anisotropic diffusion. By the help of these functions noise has been removed from the taken input that is the MRI image. These functions make it possible to recover weak signals from noise and the processed image in this way can be cleaned up without blurring or losing the clarity.

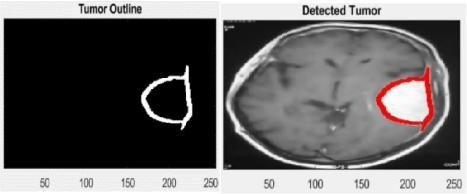
**Images Enhancement and Filtering**

In this project image improvement that is the improvement of digital image quality with none of the data concerning the first supply image degradation. The enhancement of the image starts by first converting the gray scale image to black and white image this is done using function im2bw(gray\_image)[7].Here the threshold value taken in our project is 0.6.As Image improvement strategies improve the visual look of pictures from tomography and the distinction enhancing brain volumes are linearly associated. For image sharpening the imsharpen () [7] has been used, similarly imadjust () [7] for image adjustment, freqz () for setting frequency response of image are being used. The Gaussian smoothing operator has been for the two-dimensional image convolution operators that is used to `blur' images and remove detail and noise. Gaussian is random incidence of white intensity worth, and its intensity worth is drawn from Gaussian distribution, thus it is very much use to reduce Gaussian noise and as with linear filter it is computationally economical and enhances image quality with the image boundaries for implementation of Gaussian filter the imgaussfilt()[7] has been used in our project. Color areas, which indicate the colors in an exceedingly benchmark approach by employing a reference frame and a topological space within which every color is delineated by one point of the coordinate system. The color spaces used in our image processing methods are Gray, Binary form and RGB.

**Feature Extraction**

In this phase the features of the given input image have been extracted. These features include smoothness, entropy, variance, kutosis, skewness, idm, correlation, homogeneity, mean and standard deviation . And based on these features the image is analyzed and the detection of the tumor region has been done. Below in the figure 2 there are output result of an mri image until the feature extraction phase of the project.



 (a) (b) (c) (d) (e)

(f) (g)

Figure 2:- (a)Input image, (b)Grayscale image, (c)Filtered image, (d)Bounding box, (e)Tumor alone image, (f)Tumor outline, (g)Detected tumor.

**Thresholding**

Thresholding is a type of image segmentation, where we change the pixels of an image to make the image easier to analyze. In thresholding, we convert an image from colour or grayscale into a binary image, i.e., one that is simply black and white.

**Morphological Operation**

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size.

**Code:**

**General Code:**

clc

close all

clear all

%% Input

global im im2

[path, user\_cancel] = imgetfile();

if user\_cancel

msgbox(sprintf('Invalid Selection'),'Error','Warn');

return

end

im = imread(path);

axes(handles.axes1);

imshow(im)

title('\fontsize{18}\color[rgb]{0.635 0.078 0.184} Patient''s Brain')

%% Filter

num\_iter = 10;

delta\_t = 1/7;

kappa = 15;

option = 2;

disp('Preprocessing image please wait . . .');

inp = anisodiff(s,num\_iter,delta\_t,kappa,option);

inp = uint8(inp);

inp=imresize(inp,[256,256]);

if size(inp,3)>1

inp=rgb2gray(inp);

end

figure;

imshow(inp);

title('Filtered image','FontSize',20);

%% thresholding

sout=imresize(inp,[256,256]);

t0=mean(s(:));

th=t0+((max(inp(:))+min(inp(:)))./2);

for i=1:1:size(inp,1)

for j=1:1:size(inp,2)

if inp(i,j)>th

sout(i,j)=1;

else

sout(i,j)=0;

end

end

end

%% Morphological Operation

label=bwlabel(sout);

stats=regionprops(logical(sout),'Solidity','Area','BoundingBox');

density=[stats.Solidity];

area=[stats.Area];

high\_dense\_area=density>0.7;

max\_area=max(area(high\_dense\_area));

tumor\_label=find(area==max\_area);

tumor=ismember(label,tumor\_label);

if max\_area>200

figure;

imshow(tumor)

title('tumor alone','FontSize',20);

else

h = msgbox('No Tumor!!','status');

%disp('no tumor');

return;

end

%% Bounding box

box = stats(tumor\_label);

wantedBox = box.BoundingBox;

figure

imshow(inp);

title('Bounding Box','FontSize',20);

hold on;

rectangle('Position',wantedBox,'EdgeColor','y');

hold off;

se=strel('square',5);

tumor=imdilate(tumor,se);

B=bwboundaries(tumor,'noholes');

imshow(im)

hold on

for i=1:length(B)

plot(B{i}(:,2),B{i}(:,1), 'y' ,'linewidth',1.75);

end

title('\fontsize{18}\color[rgb]{0.635 0.078 0.184} Detected Tumor')

hold off

**Gui Code:**

function varargout = testing(varargin)

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

'gui\_Singleton', gui\_Singleton, ...

'gui\_OpeningFcn', @testing\_OpeningFcn, ...

'gui\_OutputFcn', @testing\_OutputFcn, ...

'gui\_LayoutFcn', [] , ...

'gui\_Callback', []);

if nargin && ischar(varargin{1})

gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

[varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

gui\_mainfcn(gui\_State, varargin{:});

end

% End initialization code - DO NOT EDIT

% --- Executes just before testing is made visible.

function testing\_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.

function varargout = testing\_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.

% --- Executes on button press in Load\_Image.

function Load\_Image\_Callback(hObject, eventdata, handles)

global im im2

[path, user\_cancel] = imgetfile();

if user\_cancel

msgbox(sprintf('Invalid Selection'),'Error','Warn');

return

end

im = imread(path);

axes(handles.axes1);

imshow(im)

title('\fontsize{18}\color[rgb]{0.635 0.078 0.184} Patient''s Brain')

% --- Executes on button press in Output.

function Output\_Callback(hObject, eventdata, handles)

global im

axes(handles.axes2)

iterations = 10; % number of iterations.

delta\_t = 1/7; %max value of .25 for stability

kappa = 15; %conduction coefficient 20-100 ?

opt = 2; %Perona Malik diffusion equation No 2

disp('Preprocessing given input please wait . . .');

inp = anisodiff(im,iterations,delta\_t,kappa,opt); %anisotropic diffusion

inp = uint8(inp);

inp=imresize(inp,[256,256]);

if size(inp,3)>1

inp=rgb2gray(inp);

end

sout=imresize(inp,[256,256]);

t0=mean(im(:));

th=t0+((max(inp(:))+min(inp(:)))./2);

for i=1:1:size(inp,1)

for j=1:1:size(inp,2)

if inp(i,j)>th

sout(i,j)=1;

else

sout(i,j)=0;

end

end

end

label=bwlabel(sout);

stats=regionprops(logical(sout),'Solidity','Area','BoundingBox');

density=[stats.Solidity];

area=[stats.Area];

high\_dense\_area=density>0.7;

max\_area=max(area(high\_dense\_area));

tumor\_label=find(area==max\_area);

tumor=ismember(label,tumor\_label);

if max\_area>200

set(handles.edit3,'string',' Tumor!! Detected');

else

set(handles.edit3,'string',' No Tumor!! Detected');

return;

end

box = stats(tumor\_label);

wantedBox = box.BoundingBox;

hold on;

rectangle('Position',wantedBox,'EdgeColor','y');

hold off;

se=strel('square',5);

tumor=imdilate(tumor,se);

B=bwboundaries(tumor,'noholes');

imshow(im)

hold on

for i=1:length(B)

plot(B{i}(:,2),B{i}(:,1), 'y' ,'linewidth',1.75);

end

title('\fontsize{18}\color[rgb]{0.635 0.078 0.184} Detected Tumor')

hold off

function edit1\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit2\_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.

function edit2\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function Untitled\_1\_Callback(hObject, eventdata, handles)

function author\_Callback(hObject, eventdata, handles)

msgbox(sprintf('Name:Muhammad Umair Khan\n Email:FA18-bce-010@cuiatk.edu.pk'),'Author')

function edit3\_Callback(hObject, eventdata, handles)

function edit3\_CreateFcn(hObject, eventdata, handles)

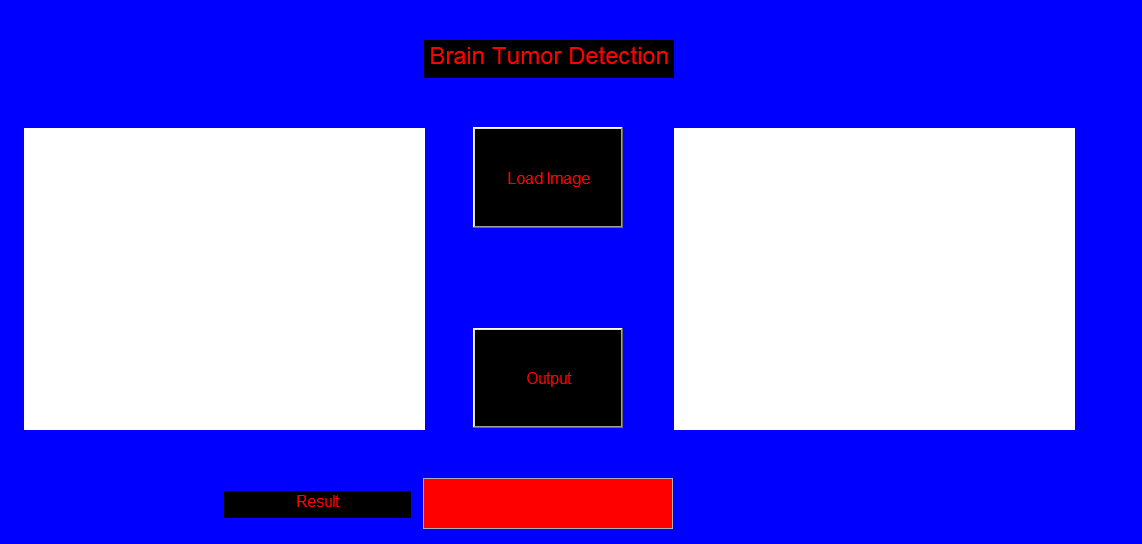
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

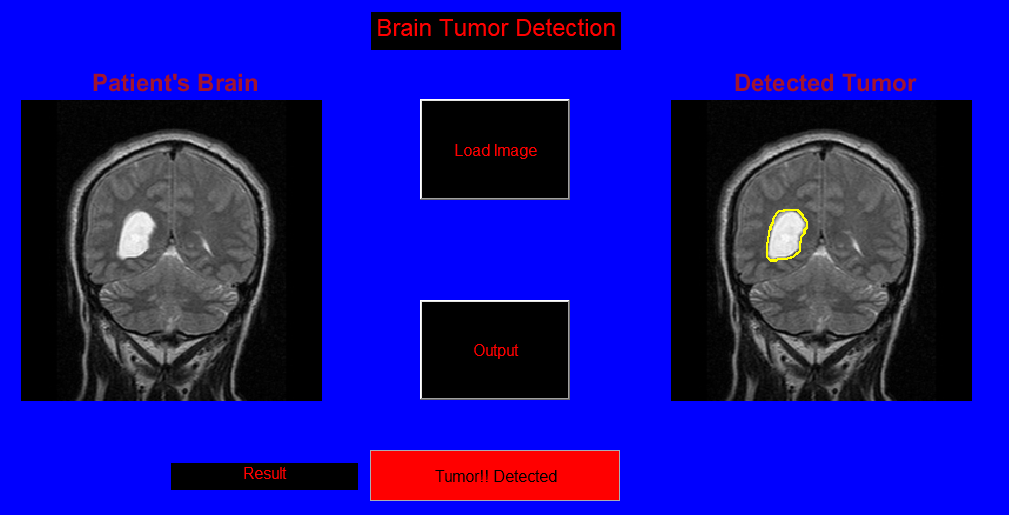
end

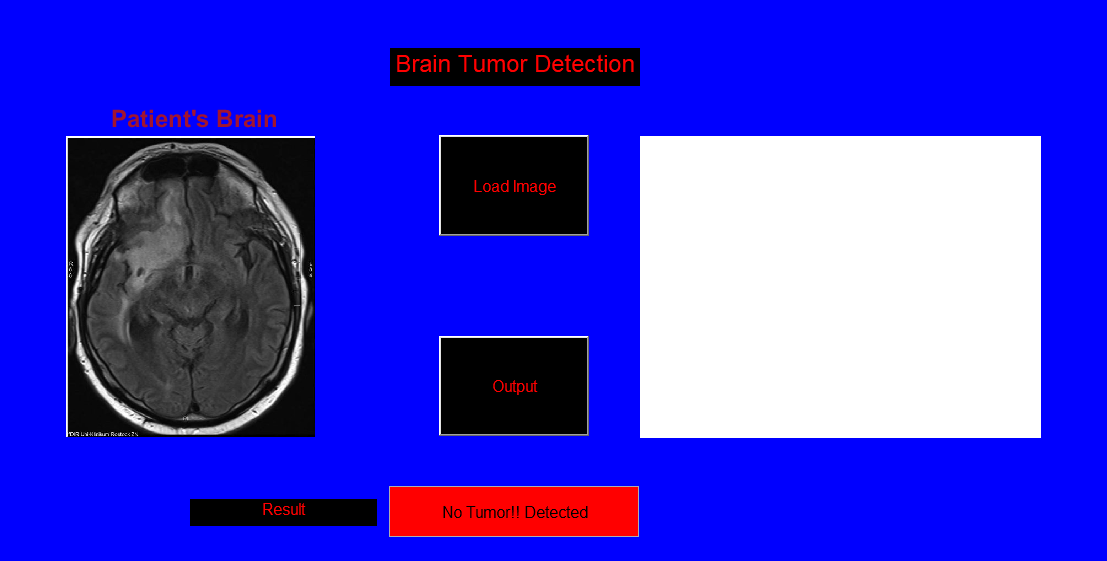
**Results:**

**Gui Structure:**

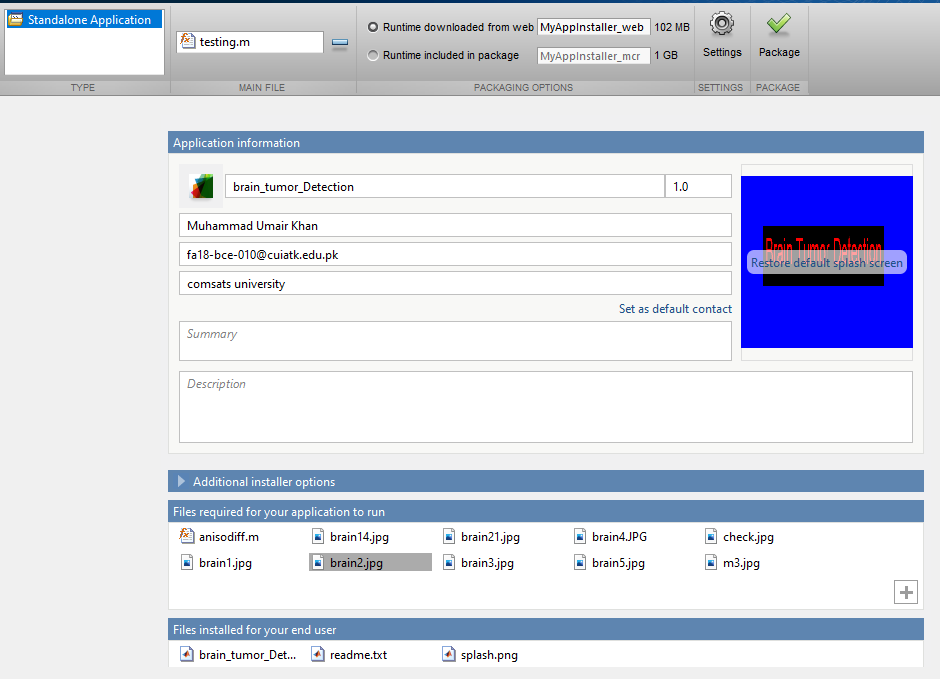


**Result for tumors:**

**Result for no tumors:**



**Application for End User**

**CONCLUSION**

In this project we have automated the diagnosis procedure for the brain tumor detection using image processing. Apart from several existing brain tumor segmentation and detection methodology are present for MRI of brain image our project has proved to provide an overall accuracy more efficiently. All the steps for detecting brain tumor that have been discussed starting from mri image acquisition ,pre-processing steps to successfully detection of the tumor using the segmentation techniques has been done.

Pre-processing involves operations like Filter based methods has been discussed. Quality enhancement and filtering are important because edge sharpening, enhancement, noise removal and undesirable background removal are improved the image quality as well as the detection procedure. Among the different filtering technique, Gaussian filter suppressed the noise without blurring the edges and it is better outlier without reducing sharpness of the images. reduces the noise; enhance the image quality and computationally more efficient than other filtering methodology. After the image quality improvement and noise reduction discussed here, segmentation methodology for a brain tumor from MRI of brain image has been used. Segmentation for segment tumor is accurately used and manufacture sensible results for big information set however undesirable behaviors can occur in case wherever a category is unrepresented data. Clustered based segmentation performs is straight forward, quick and manufacture sensible results for non-noise image except for noise pictures it leads to serious inaccuracy within the segmentation . In neural network-based segmentation perform better on noise field and no need of assumption of any fundamental data allocation but learning process is one of the great disadvantages of it.

**REFERENCE**

[1]. Sijbers J et al (1998), “Estimation of the noise in magnitude MR images”, Magn Reson Imaging 16(1):87–90.

[2]. An S, An D (1984) Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE

Trans Pattern Anal Mach Intell 6:721–741

[3]. Nowak RD (1999) Wavelet-based Rician noise removal for magnetic resonance imaging. IEEE Trans Image Process 8(10):1408–1419.

[[4]. https://www.researchgate.net/deref/http%3A%2F%2Fdicom.nema.org%2Fmedical%2Fdicom%2Fcurrent%2Foutpu](https://www.researchgate.net/deref/http%3A%2F%2Fdicom.nema.org%2Fmedical%2Fdicom%2Fcurrent%2Foutput%2Fchtml%2Fpart12%2Fchapter_K.html) [t%2Fchtml %2Fpart12%2Fchapter\_K.html.](https://www.researchgate.net/deref/http%3A%2F%2Fdicom.nema.org%2Fmedical%2Fdicom%2Fcurrent%2Foutput%2Fchtml%2Fpart12%2Fchapter_K.html)

[5]. National institute of biomedical imaging and bioengineering [(https://www.nibib.nih.gov/science-](https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri) [education/science-topics/magnetic-resonance-imaging-mri).](https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri)

[6]. Amrutha Ravi, “Medical Image Segmentation”, Computer Methods And Programs In Biomedicine 84 (2006) 63–65.

[7]. [MathWorks(https://in.mathworks.com/help/wavelet/ug/lifting-method-for-constructing-wavelets.html).](https://in.mathworks.com/help/wavelet/ug/lifting-method-for-constructing-wavelets.html)

[8]. MATLAB tool (2016)-09806-07443-53955-64350-21751-41297(URL-in.mathworks.com/products/matlab.html?s\_tid=hp\_ff\_p\_matlab)

[9]. Hava T.siegelmann,Vladimir Vapnik,David Horn ,AsaBen-Hur,”Support Vector Clustering”,Journal of Machine Learning Research 2(2001) 125-137.

[10]. Professor Teuvo Kohonen ,”Self-Organising Maps”,Springer series in Information Science,1995.

[11]. Swapnil R.Telrandhe,”segmentation methods for medical image analysis”,tesis no 1434,center for medical image science and visualization ,se-58185 linkoping ,Sweden.

[12]. Malathi Hong-Long , , “Segmentation C- Means Clustering With Spatial Information For Image Segmentation,” Computerized Medical Imaging And Graphics 30 (2006) 9–15.

[13]. Rajeshwari G tayade,Michel Crucian , “Unsupervised And Semi-Supervised Clustering: A Brief Survey,” Inria Rocquencourt, B.P. 105 78153 Le Chesnay Cedex, France.

[14]. Lukas L. et Al. Brain Tumor Classification Based On Long Echo Proton Mrs Signals. Artificial Intelligence In Medicine, 31:73–89, 2004.