

Brain Tumor Detection Using MRI Images

^{#1}Mr. Dere Avinash B., ^{#2}Mr. Patil saurabh P., ^{#3}Mr. Veer rushikesh B.,
^{#4}Mr. Saindane Nikhil A., ^{#5}Dr. Bendre Mininath R.



¹avinashdere99@gmail.com
²sppcomputer1234@gmail.com
³rushi.veer9@gmail.com
⁴nikhilsaindane53@gmail.com
⁵mininath.bendre@gmail.com

^{#12345}Department of Computer Engineering,
PREC LONI, Savitribai Phule Pune University

ABSTRACT

For brain tumor segmentation systems extraction of relevant features is having significant importance. In this paper, with the objective of improving brain tumor segmentation accuracy, we present an improved feature extraction component to take advantage of the correlation between intracranial structure deformation and the compression from brain tumor growth. Using 3-dimensional non-rigid registration and deformation modeling techniques, the component is capable of measuring lateral ventricular (LaV) deformation in the volumetric magnetic resonance (MR) images. By validating the location of the extracted LaV deformation feature data and applying the features on brain tumor segmentation with widely used classification algorithms tumor is detected.

Keywords— Feature Extraction, MRI images, Grayscale, LaV, K-Means, SVM, GLCM.

ARTICLE INFO

Article History

Received: 7th June 2019

Received in revised form :
7th June 2019

Accepted: 9th June 2019

Published online :

10th June 2019

I. INTRODUCTION

Brain tumor segmentation tries to separate healthy tissue from tumorous regions. This is an essential step in diagnosis and treatment planning in order to maximize the likelihood of successful treatment. Due to the slow and tedious nature of manual segmentation, computer algorithms that do it faster and accurately are required. Due to the inconsistent appearance and shape of a brain, segmenting brain tumors from imaging data is one of the most challenging tasks in medical image analysis.

II. LITERATURE SURVEY

Brain is a complex organ because it contains more than 10 billion working brain cells. Primary brain tumors may be either malignant (contain cancer cells) or benign (do not contain cancer cells). A primary brain tumor is a tumor which rising in the brain tissue. If a cancerous neoplasm starts elsewhere within the body, it will unfold cancer cells that grow within the brain. These kind of tumors are called secondary or metastatic brain tumors. The malignant tumor tends to grow and spread in a rapid and uncontrolled way that can cause death and the Tumor are graded according to how aggressive. They are as

- LGG: Low Grade Tumor (Benign stage)
- HGG: High Grade Tumor (Malignant stage)

A research paper by Dr.A.R.Kavitha [1], explains about using Genetic Algorithm to segment the MRI brain tumor images. Pre-processing was done using Wiener Filter (a 2D adaptive noise removal filter and it uses pixel-wise adaptive wienwer method). SVM classifier gets trained and ready for classifying test images as GLCM features are extracted from segmented and given to SVM classifier. Another analysis paper by alan jose[2] did neoplasm segmentation victimization k-mean agglomerative and fuzzy C-mean algorithm and its space calculation. After the segmentation, (which is done using k-means clustering and fuzzy c-means algorithms) the brain tumor is detected and its exact location is identified.

III. METHODOLOGY

Data set Folder Structure:

Our Training HGG –

This folder is having brain images of 220 patients. There is a different folder for each patient. There are 5 different MRI images for each Patient. The 5 different images are T1,

T2, T1C, FLAIR and OT (Ground truth of tumor Segmentation) . All these image files are stored in .mha format.

LGG - This folder contains brain images of 54 patients. There is a different folder for each patient. There are 5 different MRI images for each Patient. The 5 different images are T1,Tat2,T1C,FLAIR and OT(Ground truth of tumor Segmentation) . All these image files are stored in .mha format.

Testing

HGG LGG - This folder contains brain images of 110 patients. There is a different folder for each patient. There are 4 different MRI images for each Patient. The 4 different images are T1,T2,T1C and FLAIR. All these image files are stored in .mha format. After gaining insights in domain and lot of research, we have decided to use only FLAIR images for our work.

We started with taking each image from the training set and segment the tumor from the image using watershed clustering (region-growing) technique. The GLCM features were extracted on the segmented image. We repeat the above process for every image in the data set and use them to train our model - random forest Classifier. Then, for every testing image, we segment the tumor using the same method (Watershed) and then classify the tumor as benign or malignant using our model.

The whole project is basically divided into for major sections, namely,

- Data Processing
- Segmentation
- Feature Extraction
- Classification

Data Processing

Gaussian noise is removed from the image by using highly effective Gaussian filtering. The images we got from BRaTS were very noisy and had to be de-noised to be used for segmentation.

Gaussian Blur [4]: In this approach, a Gaussian kernel is used. It is done with the function, `cv2.GaussianBlur()`. We should mention the width and height of the kernel which should be positive and odd. We also should mention the standard deviation in the X and Y directions, `sigmaX` and `sigmaY` respectively. If only `sigmaX` is specified, `sigmaY` is taken as equal to `sigmaX`. If both are given as zeros, they are measured from the kernel size. If you want, you can create a Gaussian kernel using the function,

```
cv2.getGaussianKernel().
```

Segmentation

The data set has brain images with tumor which we segmented to provide it to the feature extraction module. After segmentation we obtained two major regions in the

image- tumor and non-tumor. The segmentation was done using Watershed Algorithm.

K-means clustering Algorithm:

K-means clustering algorithm is an unsupervised algorithm and it is used to segment (highlight) the interest area from the background. But before applying K -means algorithm, first partial stretching enhancement is enforced on the image to improve the quality of the image.

Watershed Algorithm [5]: OpenCV implemented a marker-based water- shed algorithm which is an interactive image segmentation method. It gives different labels for the object we know. Mark the region which we are sure of being the foreground or object with one color (or intensity), mark the region which we are sure of being background or non-object with another color and finally the region which we are not sure of anything, mark it with 0. That is our marker. Then apply watershed algorithm. Then our marker will be updated with the marks we gave, and the boundaries of objects will have a value of -1. This is basically a region-growing approach.

Feature Extraction

Now we need to extract some features from images as we need to do a binary classification of them using a classifier which needs these features to get trained on. We chose to extract GLCM (texture-based features).

GLCM Features [6]: After obtaining the segmented images from the above subsection, we extract GLCM features and store them. GLCM stands for Gray-Level Co-occurrence Matrix. Texture Analysis Using the Gray-Level Co-Occurrence Matrix (GLCM) is a statistical method of examining texture that considers the spatial relationship of pixels.

Classification

We used Random Forest for classification. The classifier was trained on the GLCM features acquired from the segmented images and used to classify benign (LGG) and malignant (HGG).

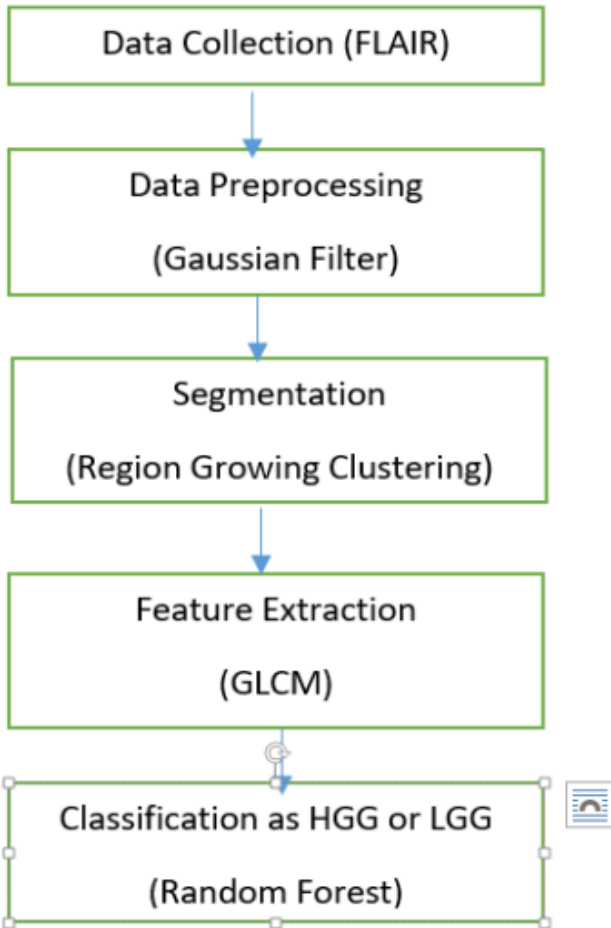
Random Forests [7]: When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample. This oob (out-of-bag) data is used to get a running impartial estimate of the classification error as trees are added to the forest. It is also used to get evaluation of variable importance. After each tree is built, all of the data are run down the tree, and proximities are computed for each pair of cases. The proximity of cases is increased by one if two cases occupy the same terminal node. At the end of the run, dividing by the number of trees the proximities are normalized. Proximities are used in locating outliers, replacing missing data, and producing illuminating low-dimensional views of the data.

Tools

- We have implemented our project in Python
- Read .mha images: SimpleITK

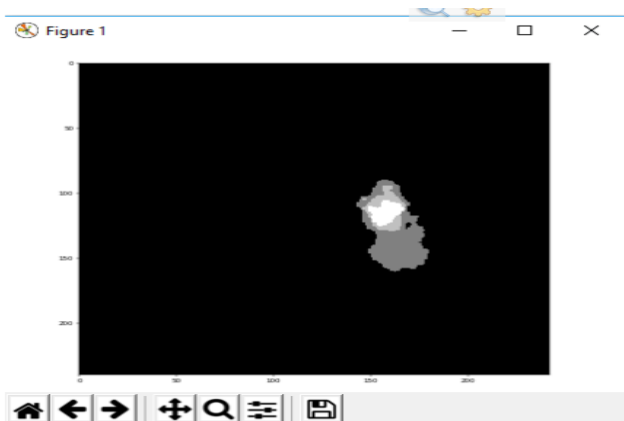
- Gaussian Blur: OpenCV
- Watershed algorithm: SciPy, scikit-image, and OpenCV
- GLCM : Package mahotas
- Random forest: RandomForestClassifier from sklearn.ensemble

IV. FLOW CHART



V. RESULT AND DISCUSSION

Segmented Image: Input image is segmented using K-mean



Predicting the Tumor: The Segmented images are given as training dataset. A model/Classifier is created using Rainforest. 8 new images are tested and output is predicted with 87.5% accuracy.

	Linear SVM	Rainforest
Accuracy	0.75	0.875
Precision	[0.75,0.75]	[1.0,0.8]
Recall	[0.75,0.75]	[0.75,1.0]
Fscore	[0.75,0.75]	[0.857,0.888]
Support	[4,4]	[4,4]

Table: Accuracy comparison between linear SVM and Rainforest.

```

Training labels: (100,)
[STATUS] Creating the classifier..
[STATUS] Fitting data/label to model..
Test\HGG\brats_tc1a_pat179_0001\VSD.Brain_3more.XX.0.OT.42375\VSD.Brain_3more.XX.0.OT.42375.mha
HGG
Test\HGG\brats_tc1a_pat180_0001\VSD.Brain_3more.XX.0.OT.42379\VSD.Brain_3more.XX.0.OT.42379.mha
LGG
Test\HGG\brats_tc1a_pat183_0001\VSD.Brain_3more.XX.0.OT.42381\VSD.Brain_3more.XX.0.OT.42381.mha
HGG
Test\HGG\brats_tc1a_pat184_0001\VSD.Brain_3more.XX.0.OT.42383\VSD.Brain_3more.XX.0.OT.42383.mha
HGG
Test\LGG\brats_tc1a_pat480_0001\VSD.Brain_3more.XX.0.OT.42819\VSD.Brain_3more.XX.0.OT.42819.mha
LGG
Test\LGG\brats_tc1a_pat483_0001\VSD.Brain_3more.XX.0.OT.42821\VSD.Brain_3more.XX.0.OT.42821.mha
LGG
Test\LGG\brats_tc1a_pat490_0001\VSD.Brain_3more.XX.0.OT.42823\VSD.Brain_3more.XX.0.OT.42823.mha
LGG
Test\LGG\brats_tc1a_pat493_0001\VSD.Brain_3more.XX.0.OT.42827\VSD.Brain_3more.XX.0.OT.42827.mha
LGG
Actual:
['HGG', 'HGG', 'HGG', 'HGG', 'LGG', 'LGG', 'LGG', 'LGG']
Prediction: ['HGG', 'LGG', 'HGG', 'HGG', 'LGG', 'LGG', 'LGG', 'LGG']
Accuracy:
0.875
precision: [1. 0.8]
recall: [0.75 1. ]
fscore: [0.85714286 0.88888889]
support: [4 4]
  
```

VI. CONCLUSION

Since, we have a tendency to mechanically segmenting the growth and classify its as benign or malignant, it considerably reduces the quantity of your time that a doctor takes to spot and classify the growth. It also to some extent reduces the error made while segmenting and classifying tumor.

REFERENCES

- [1] Link for Data Set
- [2] "Brain Tumor Segmentation using Genetic Algorithm with SVM Classifier" by Dr.A.R. Kavitha, L.Chitra, R.kanaga
- [3] "Brain Tumor Segmentation Using K-Means Clustering And Fuzzy C-Means Algorithms And Its Area Calculation" by Alan Jose, S.Ravi, M.Sambath.
- [4] https://docs.opencv.org/3.1.0/d3/db4/tutorial_py_watershed.html
- [5] <https://www.smir.ch/>
- [6] https://in.mathworks.com/help/images/texture_analysis_using_the_gray_level_co_occurrence_matrix_glcm.html
- [7] https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#overview
- [8] <https://pyscience.wordpress.com/2014/10/19/image-segmentation-with-python-and-simpleitk>