Automatic Tumor Classification of Brain MRI Images

V.Vani[1*]

Department of computer science,
JJ College of arts and science, Pudukkottai

M. Kalaiselvi Geetha [2]

Department of computer science and Engineering,
Annamalai University, Tamil Nadu

Abstract — Brain tumor classification is an active research area in medical image processing and pattern recognition. Brain tumor is an abnormal mass of tissue in which some cells grow and multiply uncontrollably, apparently unregulated by the mechanisms that control normal cells. The growth of a tumor takes up space within the skull and interferes with normal brain activity. The detection of the tumor is very important in earlier stages. Automating this process is a challenging task because of the high diversity in the appearance of tumor tissues among different patients and in many cases similarity with the normal tissues. This paper depicts a novel framework for brain tumor classification based on Gray Level Co-occurrence matrix (GLCM) statistical features are extracted from the brain MRI images, which signify the important texture features of tumor tissue. The experiments are carried out using BRATS dataset, considering two classes viz (normal and abnormal) and the extracted features are modeled by Support Vector Machines (SVM), k-Nearest Neighbor (k-NN) and Decision Tree(DT)for classifying tumor types. In the experimental results, Decision Tree exhibit effectiveness of the proposed method with an overall accuracy rate of 98.68%, this outperforms the SVM and k-NN classifiers.

Keywords—MRI, GLCM, SVM, K-NN, DT, Brain Tumor, Tumor detection, BRATS

1. Introduction

Magnetic resonance images (MRI) is an advanced medical imaging technique primarily used in radiology to visualize high resolution images of the parts, structure and functions of the body. It provides detailed images of the body in any plane. MRI, scientists can visualize both surface and deep structures with a high degree of anatomical detail, and they can detect the occurrence of minute changes in these structures over time. In the earliest days, the technique was referred to as nuclear magnetic resonance imaging (NMRI). However, as the word nuclear was associated in the public mind as

Ionizing radiation exposure it is now simply referred to as MRI. MR images can also be used to track the size of a brain tumor as it responds (or doesn't) to treatment. A reliable method for classifying the tumor would clearly be a useful tool. MRI scan can be used as an accurate method for detecting tumor from human brain. Fig. 1 shows the MRI (Magnetic resonance imaging) of the human brain. Classification of tumors in magnetic resonance images (MRI) is an important task. But it is quite time consuming when performed manually by experts.
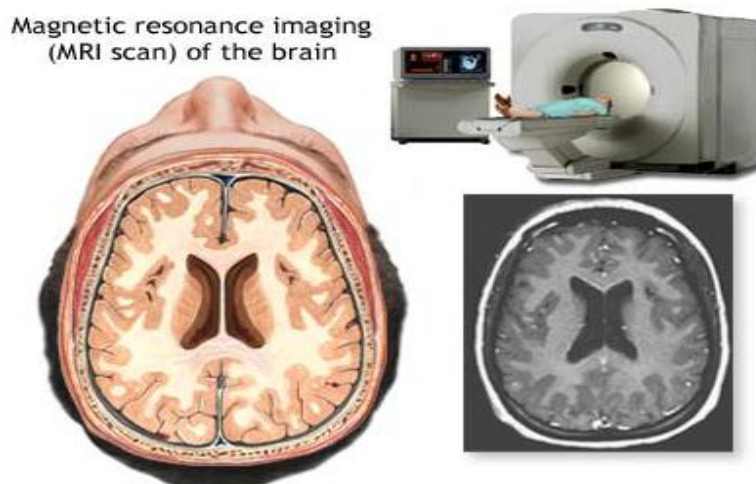

Fig. 1: A Magnetic Resonance Imaging (MRI) of the brain

Projection images are useful in determining the primary location of tumors. Automating process is challenging task due to the high diversity in appearance of tumor tissue in different patients, and in many cases, similarity between tumor and normal tissues. The images are in a standard format usable in digital imaging and communication for medicine (DICOM). This is the standard format for all medical images. It was developed by the National Electronic Manufactures Association (NEMA). This standard format is mainly used for storing, printing and transmitting information in medical imaging. Many diagnostic imaging techniques can be performed for early detection of brain tumors such as Computed Tomography (CT), Positron Emission Tomography (PET) and Magnetic Resonance Imaging (MRI). Compared to all other imaging techniques, MRI is more efficient in brain tumor detection and identification, mainly due to the high contrast of soft tissues, high spatial resolution and since it does not produce any harmful radiation, and is a non invasive technique. Fig. 2(a) and (b) shows the Magnetic Resonance Image (MRI) from BRATS database is categorized into two distinct classes as normal, abnormal brain tumor.

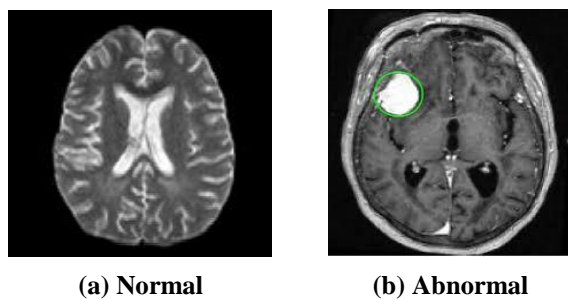**(a) Normal**   **(b) Abnormal**

Fig. 2: MRI of the normal and abnormal images of the brain

1.1 Outline of the work

This paper deals with brain tumor classification, which aims to identify the brain tumor types as normal or abnormal from the brain MRI images. The proposed approach is evaluated using BRATS 2014 dataset. Thus, the GLCM features are extracted from the MRI image as a feature set. The extracted features are modeled by SVM, k-NN and Decision tree classifiers for training and testing. The rest of the paper is structured as follows. Section 2 reviews related work. Section 3 provides an overview of the proposed approach. Section 4 describes the proposed feature extraction method and experimental results evaluating its performance on BRATS dataset are presented in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

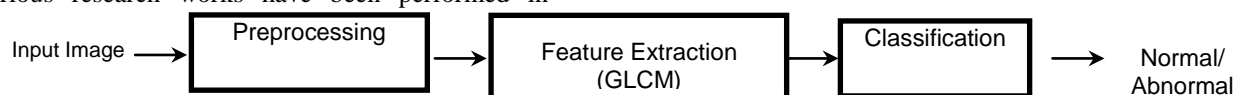From the literature survey, initially, it can be concluded that, various research works have been performed in classifying MR brain images into normal and abnormal [1], [2]. Priyanka, BalwinderSingh [3] focused on survey of well-known brain tumor detection algorithms that have been proposed so far to detect the location of the tumor. The main concentration is on those techniques which use image segmentation to detect brain tumor. Image segmentation is the process of partitioning a digital image into multiple segments. R. J. Ramteke, KhachaneMonali Y [4] proposed a method for automatic classification of medical images in two classes Normal and Abnormal based on image features and automatic abnormality detection. KNN classifier is used for classifying image. K-Nearest Neighbour (K-NN) classification technique is the simplest technique conceptually and computationally that provides good classification accuracy. Khushboo Singh, SatyaVerma [5] proposed sophisticated classification techniques based on Support Vector Machines (SVM) are proposed and applied to brain image classification using features derived. Shweta Jain [6] classifies the type of tumor using Artificial Neural Network (ANN) in MRI images of different patients with Astrocytoma type of brain tumor. The extraction of texture features in the detected tumor has been achieved by using Gray Level Co-occurrence Matrix (GLCM). Statistical texture analysis techniques are constantly being refined by researchers and the range of applications is increasing [7], [8], [9]. Gray level co-occurrence matrix method is considered to be one of the important texture analysis techniques used for obtaining statistical properties for further classification, which is employed in this research work. Probabilistic Neural Network is found to be superior over other conventional neural networks such as Support Vector Machine and Back propagation Neural Network in terms of its accuracy in classifying brain tumors [10]. Hence a wavelet and co occurrence matrix method based texture feature extraction and Probabilistic Neural Network for classification has been used in this method of brain tumor classification.

3. Proposed Approach

The general overview of the proposed approach is illustrated in Fig. 3. This approach uses the standard benchmark Brain Research and Analysis in Tissues (BRATS) tumor dataset [11] for the experiments. The input tumor images are smoothed by median filter. It is necessary to pre-process all the tumor images for robust feature extraction and classification. Then BRATS dataset divided into two classes (normal and abnormal) for feature extraction process. The extracted features are modeled using SVM, k-NN and Decision tree for classification.

Fig. 3: Block diagram of the Proposed Approach

4. Feature Extraction

The extraction of discriminative feature is most essential and vital problem with brain tumor classification, which represents the meaningful information that is vital for further study. The ensuing sections present, the representation of the feature extraction method used in this work.

4.1. Gray Level Co-occurrence Matrix(**GLCM) for Tumor Classification**

Texture analysis aims at finding a distinctive way of representing the essential characteristics of textures and represents them in simpler and unique form so that they can be used for robust, accurate recognition. A geometric method of reviewing texture that deals with the spatial connection of pixels is the gray level co-occurrence matrix. The approach and performance behind the Gray Level Co-occurrence Matrix (GLCM) method are presented in [12]. GLCM is obtained by calculating how often a pixel with grayscale intensity values i occurs adjacent to a pixel with the value j. Each element (i, j) in GLCM specifies the number of times that the pixel with the value i occurred adjacent to a pixel with value j. GLCM texture indicates the relationship between the reference and neighbour pixel of the gray level image at the various directions.
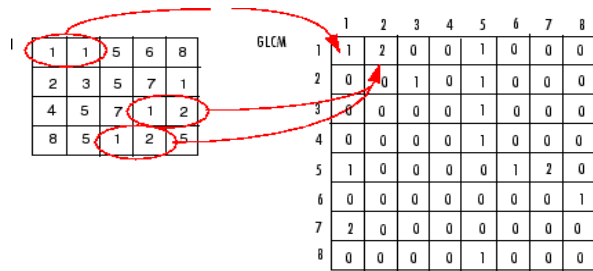

Fig. 4: GLCM Matrix for distance 1 and 0° direction.

The adjacency can be defined to take place in each of four directions 0°, 45°, 90° and 135° degrees in a two-dimensional pixel image (horizontal, vertical, left and right diagonal). GLCM matrix stores the instance occurrences between adjacent pixels. Element (1, 2) in the GLCM accommodates the value 2 because there are two occurrences of (1, 2) in the image as shown in Fig. 4. Given an Image I, of size N × N, the co-occurrence matrix P can be defined as

$$P(i,j) = \sum_{x=1}^{G} \sum_{y=1}^{G} \begin{cases} 1, & if \ I(x,y)=i \ and \ I(x+\Delta_x, y+\Delta_y)=j \\ 0, & Otherwise \end{cases}$$ (2.1)

where the offset (Δx, Δy), specifies the ranges between the pixel of interest and its neighbour. i, j specifies intensity values of the image and x, y are the spatial location in the image I. In GLCM method, 8 texture descriptors are used namely contrast, dissimilarity, entropy, sum of square variance, sum of average, sum variance, difference

variance and difference entropy. The texture descriptors are explained as follows:

Contrast:

Measure of contrast or local intensity variation can favor contributions from p(i, j) away from the diagonal, i.e. $i \neq j$

$$Contrast = \sum_{i,j=0}^{G-1} (i-j)^2 \, p(i,j)$$ (1)

Dissimilarity:

Similar to GLCM contrast and it is high if the local region has a high contract.

$$Dissimilarity = \sum_{i,j=0}^{G-1} |i-j| \, P(i,j)$$ (2)

Entropy:

This measures the randomness of the intensity distribution.

$$Entropy = \sum_{i,j=0}^{G-1} p(i,j) \, (-Inp(i,j))$$ (3)

Sum of Square Variance:

$$Variance = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i-\mu)^2 \, p(i,j)$$ (4)

Sum Average:

$$Sum \ Average = \sum_{i=0}^{2G-2} i \, P_{x+y}(i)$$ (5)

Sum Variance:

$$Sum \ Variance = \sum_{i=0}^{2G-2} (i-aver)^2 \, P_{x+y}(i)$$ (6)

Difference Variance:

$$Difference \ Variance = \sum_{i=0}^{G-1} (i-aver)^2 \, P_{x+y}(i)$$ (7)

Difference Entropy:

$$Difference\ Entropy = \sum_{i=0}^{G-1} p_{x+y}(i) \log(p_{x+y}(i))$$

(8)   (

4.2 Support Vector Machine for Classification

Support Vector Machine (SVM) [13] is based on the principle of structural risk minimization (SRM). Support vector machines can be used for pattern classification and nonlinear regression. It constructs a linear model to estimate the decision function using non-linear class boundaries based on support vectors. If the data are linearly separable, SVM trains linear machines for an optimal hyperplane that separates the data without error and into the maximum distance between the hyperplane and the closest training points. The training points that are closest to the optimal separating hyperplane are called support vectors. Fig. 5 shows the architecture of SVM. SVM maps the input patterns into a higher dimensional feature space through some nonlinear mapping chosen a priori. A linear decision surface is then constructed in this high

dimensional feature space. Thus, SVM is a linear classifier in the parameter space, but it becomes a nonlinear classifier as a result of the nonlinear mapping of the space of the input patterns into the high dimensional feature space.

4.2.1 SVM Principle:

Support vector machine (SVM) can be used for classifying the obtained data [14]. SVM are a set of related supervised learning methods used for classification and regression and they belong to a family of generalized linear classifiers. A feature vector (termed as pattern) is denoted by $x=(x_1, x_2, \ldots, x_n)$ and its class label by y such that $y = \{+1, -1\}$. Therefore, consider the problem of separating the set of n-training patterns belonging to two classes,

$$(x_i, y_i), \ x_i \in R^n, \ y = \{+1, -1\}, \ i = 1, 2, \ldots, n \quad (9)$$

A decision function g(x) can correctly classify an input pattern x that is not necessarily from the training set.
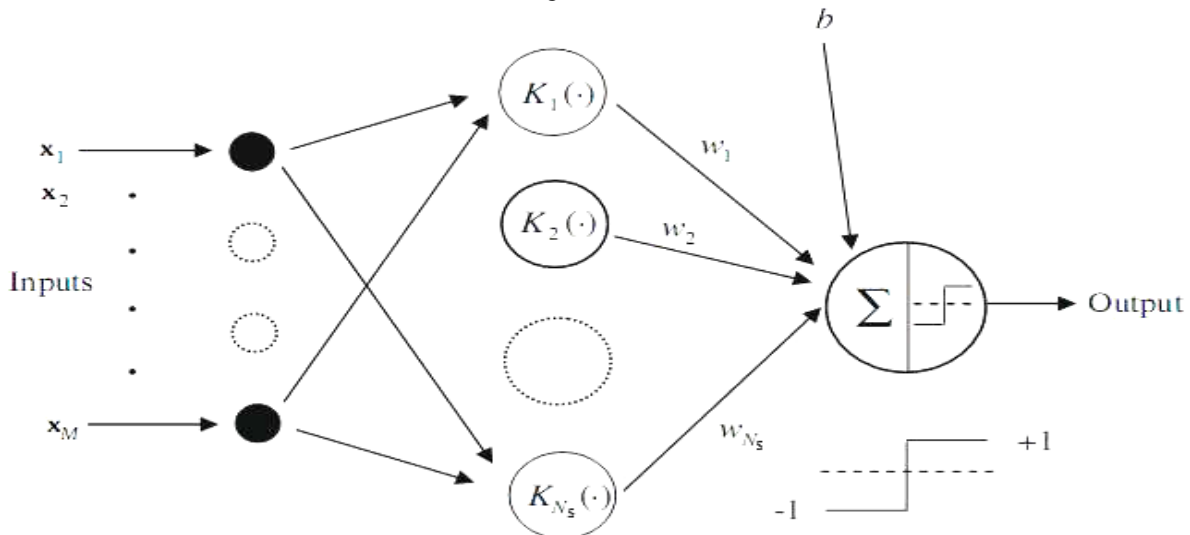

Fig. 5: Architecture of the SVM (N_s is the number of support vectors).

4.2.2 SVM for Linearly Separable Data

A linear SVM is used to classify data sets which are linearly separable. The SVM linear classifier tries to maximize the margin between the separating hyperplane and the patterns lying on the maximal margins called support vectors. Such a hyperplane with maximum margin is called maximum margin hyperplane [14]. In case of linear SVM, the discriminant function is of the form:

$$g\ (x) = w^t x + b \quad (10)$$

such that $g(x_i) \geq 0$ for $y_i = +1$ and $g(x_i) < 0$ for $y_i = -1$. In other words, training samples from the two different classes are separated by the hyperplane $g(x) = w^t x + b = 0$. SVM finds the hyperplane that causes the largest separation between the decision function values from the two classes. Now the total width between two margins is $2/w^t w$, which is to be maximized. Mathematically, this

hyperplane can be found by minimizing the following cost function:

$$J(w) = \frac{1}{2} w^t w \quad (11)$$

Subject to separability constraints

$$g(x_i) \geq +1 \ for \ y_i = +1$$
$$or$$
$$g(x_i) \leq -1 \ for \ y_i = -1 \quad (12)$$

Equivalently, these constraints can be re-written more compactly as

$$y_i(w^t x_i + b) \geq 1; \ i = 1, 2, ..., n \qquad (13)$$

For the linearly separable case, the decision rules defined by an optimal hyperplane separating the binary decision classes are given in the following equation in terms of the support vectors:

$$Y = sign\left(\sum_{i=1}^{i=N_s} y_i \alpha_i (x, x_i) + b \right) \qquad (14)$$

where Y is the outcome, y_i is the class value of the training example x_i, and represents the inner product. The vector corresponds to an input and the vectors x_i, i = 1, . . . ,N_s, are the support vectors. In Eq. (14), b and α_i are parameters that determine the hyperplane.

4.2.3 SVM for linearly non-separable data:

For non-linearly separable data, it maps the data in the input space into a high dimension space $x \in \Box^I \rightarrow \Phi(x) \in \Box^H$ with kernel function $\Phi(x)$, to find the separating hyperplane.

4.2.4 Determining support vectors:

The support vectors are the (transformed) training patterns. The support vectors are (equally) close to hyperplane. The support vectors are training samples that define the optimal separating hyperplane and are the most difficult patterns to classify. Informally speaking, they are the patterns most informative for the classification task.

4.2.5 Inner product kernels:

SVM generally applies to linear boundaries. If a linear boundary is inappropriate, SVM can map the input vector into a high dimensional feature space. By choosing a non-linear mapping, the SVM constructs an optimal separating hyperplane in this higher dimensional space. The function K is defined as the kernel function for generating the inner products to construct machines with different types of non-linear decision surfaces in the input space.

$$(X, x_i) = \Phi(X). \ \Phi(x_i) \qquad (15)$$

The kernel function may be any of the symmetric functions. There are several SVM kernel functions as given in Table 1.

Table 1: Types of SVM inner product kernels.

Types of kernels	Inner Product Kernel	Details
Polynomial	$(x^T x_i + 1)^p$	Where x is input patterns, x_i is support vectors, σ^2 is variance, $1 \leq i \leq N_s$,
Gaussian	$exp\left[-\dfrac{\|x^T - x_i\|^2}{2\sigma^2} \right]$	
Sigmoidal	$tanh(\beta_0 x^T x_i + \beta_1)$	N_s is number of support vectors, β_0, β_1 are constant values. p is degree of the polynomial

The dimension of the feature space vector $\Phi(x)$ for the polynomial kernel of degree p and for the input pattern dimension of d is given by

$$\frac{(p+d)!}{p!d!} \qquad (16)$$

For sigmoidal kernel and Gaussian kernel, the dimension of feature space vectors is shown to be infinite. Finding a suitable kernel for a given task is an open research problem. Given a set of images corresponding to N subjects for training, N SVMs are trained. Each SVM is trained to distinguish between all images of a single person and all other images in the training set. During testing, the class label l of a face pattern x can be determined using (17)

$$l = \begin{cases} n, \ if \ d_n(x) + t > 0 \\ 0, \ if \ d_n(x) + t \leq 0 \end{cases} \qquad (17)$$

where, $d_n(x) = \max\{d_i(x)\}_{i=1}^N$ and $d_i(x)$ is the distance from x to the SVM hyperplane corresponding to person i. The classification threshold is t, and the class label l = 0 stands for unknown.

4.3 k-Nearest Neighbour for Classification

The k-NN classifier ranks the test formula's neighbors among the training vectors and uses the category labels of the k most similar neighbors to predict categories of the test formula [15], [16]. In traditional k-NN, the value k is fixed and usually determined experimentally. If the k is too large, big classes (a lot of members in the class) may dominate small ones. Incorrect categories may be assigned for multi-label classification. In the opposite, if k is too small, the advantages of this algorithm to make use of many experts will not be presented. Moreover, in multi-label classification, the test formula may not be assigned to all categories. It should be in k-NN algorithm, the most popular on similarity, i.e., cosine similarity, which can be calculated by the dot product between these two vectors. In case both vectors are normalized into the unit length, the value of similarity of the two vectors is in the range of 0 and 1.

$$C(f_t) = \arg \max \sum_{f_j \in kNN} z(f_j, c_k) \qquad (18)$$

When the k nearest neighbors is set, several strategies could be taken to predict the category of a test

formula. Two strategies that are widely used are listed as follows. Where f_l is a test formula f_j is one of the neighbors (k-NN) in the training set, $z(f_j, c_k)$. 0,1 indicates whether f_j belongs to class c_k in the set of classes C, and sim (f_l, f_j) is the similarity function between f_l and f_j . For single-label classification, the above equation means that the prediction will be a category that has the largest number of members in the k nearest neighbors. The Eq. (19) expresses that the category which has maximal sum of similarity (score), will be assigned. This strategy is thought to be more useful and is more widely used.

$$C(f_t) = \arg\max_{ck \in C} \sum_{f_j \in kNN} sim(f_l, f_j) \, z(f_j, c_k)$$

(19)

4.4 Decision Tree for Classification

Decision tree is one of the preparatory learning algorithms that construct a classification tree to classify the data [17] and decision tree represents rules. The classification tree is made by recursive partitioning of feature space based on a training set. A decision tree is visual representation of a problem. A decision tree helps to decompose a complex problem into smaller and more manageable undertakings. Decision tree is a common and intuitive approach to classify a pattern through sequence of questions in which the next question depends upon the answer to current question. Decision tree analysis is a formal, structured approach to make decisions. It is based on the "divide and conquer" strategy.

There are two common issues for construction of decision trees [18]:
   (a) Growth of the tree to accurately categorize the training dataset, and
   (b) The pruning stage, whereby superfluous nodes and branches are removed in order to improve classification accuracy.
A decision tree is in the form of a tree structure, where

each node is either:
   1.   A leaf node - indicates the value of the target class of examples, or
   2.   A decision node - specifies some test to be carried out on a single attribute-value, with two or more than two branches and each branch has a sub-tree.
Decision trees are the commonly used method for pattern classification.

Decision tree is a common and intuitive approach to classify a pattern through sequence of questions in which the next question depends upon the answer to the current question. A decision tree is a visual representation of a problem. A decision tree helps to decompose a complex problem into smaller, more manageable undertakings. This allows the decision makers to make smaller determinations along the way to achieve the optimal overall decision. Decision tree analysis is a formal, structured approach to make decisions.

5. Experimental Results

In this section, the proposed method is evaluated using BRATS tumor dataset. The experiments are carried out in MATLAB 2013a in Windows 7 Operating System on a computer with Intel Xeon Processor 2.40 GHz with 4 GB RAM. The obtained GLCM features are fed to supervised classifiers such as SVM, K-NN and Decision tree to develop the model for each class, and these models are used to test the performance of the proposed features.

5.1 BRATS Dataset

Multimodal Brain Tumor Image Segmentation (BRATS) is a large dataset of brain tumor MR scans in which the relevant tumor structures have been delineated.   In this work, 200 images are taken for evaluation. For conducting the experiments, 120 images are taken as training samples and the remaining 80 images are considered for testing
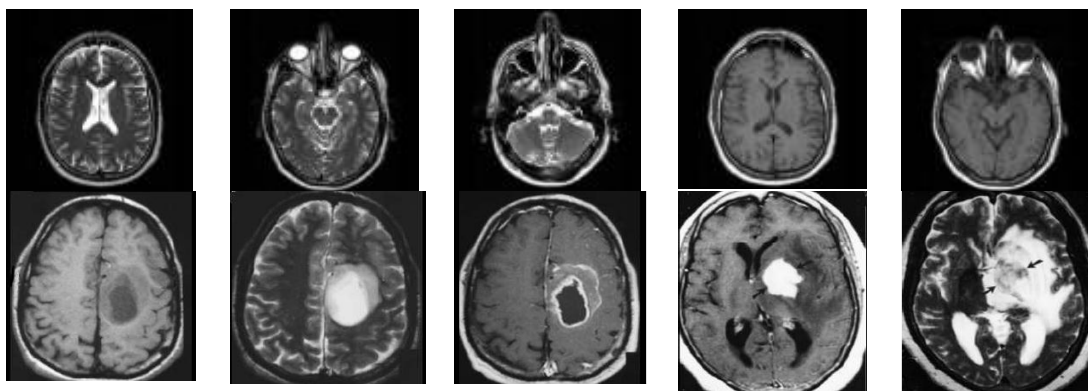

Fig. 6: Sample brain MRI images of the BRATS dataset: Normal (top row) and Abnormal (bottom row)

5.2 Quantitative Evaluation

An efficient study of performance measure for classification tasks is presented in [19]. Precision (P), Recall (R) and F-measure (F) are the commonly used evaluation metrics and these measures are used to evaluate the performance of the proposed method. These measures provide the best perspective on classifiers performance for classification. Table 2 shows confusion matrix for classification.

	Predicted Outcomes	
	Positive	Negative
Positive`	TP	FN
Negative`	FP	TN

Table 2: Confusion matrix for classification.

The confusion matrix contains information about actual and predicted classifications done by a classification system, where, TP and TN are the number of true positive and true negative predictions for the particular class. FN and FP are the number of false negative and false positive for the particular class. The classification widely uses Precision, Recall and F-measure, which do not detect changes in TN when all other matrix entries remain the same. The precision (P) is calculated as in (20). The Recall (R) or Sensitivity is calculated as in (21).

$$\text{Precision (P)} = \frac{TP}{TP+FP} \quad (20)$$

$$\text{Recall (R)} = \frac{TP}{TP+FN} \quad (21)$$

$$\text{F-Measure (F)} = 2 \times \frac{P \times R}{P+R} \quad (22)$$

$$\text{Accuracy (A)} = \frac{TP+TN}{TP+FP+TN+FN} \quad (23)$$

Precision and Recall do not depend on TN, but only on the correct labeling of positive examples (TP) and the incorrect labeling of examples (FP and FN). These measures provide the most excellent perspective on classifier performance for brain tumor classification. The F-measure is a combined measure of precision and recall metrics and it is calculated as in (22). The Accuracy is calculated as in (23).

5.3 Results obtained with SVM The confusion matrices of the SVM classifier on BRATS dataset is shown in Table 3, where diagonal of the table shows that accurate responses of tumor types. The average recognition rate of SVM is 95.94%. In SVM, the normal class is almost classified well, where as in abnormal class is confused

with normal class as 8.11%. Thus, it needs further attention.

Table 3: Confusion matrix for SVM

	Normal	**Abnormal**
Normal	**100**	0.0
Abnormal	8.11	**91.89**

5.4 Results obtained with k-NN

The confusion matrices of the k-NN classifier on BRATS dataset is shown in Table 4, where diagonal of the table shows that accurate responses of tumor types. The average recognition rate of k-NN is 77.03%. In k-NN, the normal class is classified well and good, where as the abnormal class is confused with normal class as 45.95%.

Table 4: Confusion matrix for KNN

	Normal	**Abnormal**
Normal	**100**	0.0
Abnormal	45.95	**54.05**

5.5 Results obtained with Decision Tree

The confusion matrices of the Decision Tree classifier on BRATS dataset is shown in Table 5, where diagonal of the table shows that accurate responses of tumor types. The average recognition rate of DT is 98.68%. In DT, the normal class is classified well, where as the abnormal class minute confused with normal class as 2.63%.

Table 5: Confusion matrix for Decision Tree

	Normal	**Abnormal**
Normal	**100**	0.0
Abnormal	2.63	**97.37**

The quantitative evaluation results are tabulated in Table 6, which shows that the proposed approach has a higher precision, recall and F-measure for the Decision tree classifier on BRATS dataset, when compared to SVM and k-NN classifiers. The overall performance of the proposed method with various classifiers on BRATS dataset is shown in Fig. 7.

Table 6: Performance measure of the BRATS dataset on SVM, k-NN and DT classifiers

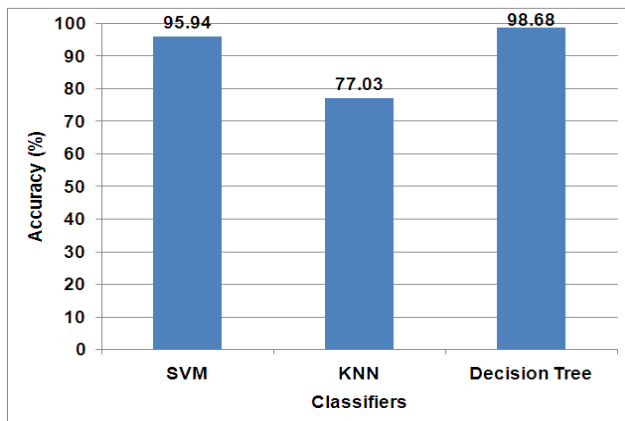| Classifiers	Precision	Recall	F-measure
SVM	94.00	95.95	94.70
K-NN	78.21	77.03	71.15
Decision Tree	**97.83**	**98.68**	**98.22**

Fig 7: Overall accuracy obtained for BRATS dataset on SVM, k-NN and DT classifiers

6. Conclusion and Future Work

This paper presents an efficient method of classifying MR brain images into normal and abnormal tumor, using a SVM, k-NN and Decision Tree. This paper presents a method called Gray Level Co-occurrence matrix (GLCM) statistical features is extracted from the brain MRI images, which signify the important texture features of tumor tissue and gives very promising results in classifying MR images. From the experimental results, it is observed that Decision Tree shows a classification accuracy of 98.68%, and demonstrated that the proposed feature method performs well and achieved good recognition results for tumor classification. It is observed from the experiments that the system could not distinguish abnormal class with high accuracy and is of future interest.

References

[1] Ahmed kharrat, Karim Gasmi, et.al, "A Hybrid Approach for Automatic Classification of Brain MRI Using Genetic Algorithm and Support Vector Machine," Leonardo Journal of Sciences, pp.71-82, 2010.

[2] Ahmed Kharrat, Mohamed Ben Messaoud, et.al, "Detection of Brain Tumor in Medical Images," International Conference on Signals, Circuits and Systems IEEE, pp.1-6, 2009.

[3] Priyanka, Balwinder Singh. "A review on brain tumor detection using segmentation." International Journal of Computer Science and Mobile Computing (IJCSMC) 2.7 (2013): 48-54.

[4] Ramteke, R. J., and Y. Khachane Monali. "Automatic medical image classification and abnormality detection using K-Nearest Neighbour." International Journal of Advanced Computer Research 2.4 (2012): 190-196.

[5] Vishnumurthy T D, Mohana H S Vaibhav A Meshram and Pramod Kammar, "Suppression of Herringbone Artifact in MR Images of Brain Using Combined Wavelet and FFT Based Filtering Technique", International Journal of Computer Sciences and Engineering, Volume-04, Issue-02, Page No (66-71), Feb -2016

[6] Azzeddine Riahi, "Image Segmentation Techniques Based on Fuzzy C-Means and Otsu, Applied to the Brain MRI in Tumor Detection", International Journal of Computer Sciences and Engineering, Volume-03, Issue-12, Page No (89-101), Dec -2015

[7] Qurat-Ul-Ain, Ghazanfar Latif, "Classification and Segmentation of Brain Tumor using Texture Analysis," Recent Advances In Artificial Intelligence, Knowledge Engineering And Data Bases, pp 147-155, 2010.

[8] Vipin Y. Borole, Seema S. Kawathekar, "Study of various DIP Techniques used for Brain Tumor detection and tumor area calculation using MRI images", International Journal of Computer Sciences and Engineering, Volume-04, Issue-07, Page No (39-43), Jul -2016,

[9] G Vijay Kumar and G V Raju, "A Real-Time Approach to Brain Tumor Detection Implementing Wavelets and ANN", International Journal of Computer Sciences and Engineering, Volume-03, Issue-11, Page No (89-93), Nov -2015

[10] Parag P. Bharne and Deepak Kapgate, "A Review of Classification Techniques in Brain Computer Interface", International Journal of Computer Sciences and Engineering, Volume-02, Issue-12, Page No (68-72), Dec -2014.

[11] Menze, Bjoern H., et al. "The multimodal brain tumor image segmentation benchmark (BRATS)." IEEE Transactions on Medical Imaging 34.10 (2015): 1993-2024.

[12] F. Albregtsen, "Statistical texture measures computed from GLCM", Image processing Laboratory, Dept of Informatics, University of Oslo, 2008.

[13] Nello Cristianini and John Shawe-Taylor, An introduction to support vector machines and other kernel-based learning methods, Cambridge university press, 2000.

[14] Vladimir Naumovich Vapnik and Vlamimir Vapnik, Statistical learning theory, vol. 1, Wiley New York, 1998.

[15] Keller, James M., Michael R. Gray, and James A. Givens. "A fuzzy k-nearest neighbor algorithm." IEEE transactions on systems, man, and cybernetics 4 (1985): 580-585.

[16] N. Bhatia et al, Survey of Nearest Neighbor Techniques. International Journal of Computer Science and Information Security, Vol. 8, No. 2, 2010.

[17] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen, Classification and regression trees, CRC press, 1984.

[18] J. Ross Quinlan, "Induction of decision trees," Machine learning, vol. 1, no. 1, pp. 81–106, 1986.

[19] Geetika Gupta, RupinderKaur, ArunBansal, MunishBans al, "Analysis and Comparison of Brain Tumor Detection and Extraction Techniques from MRI Images." International Journal of Advanced Research in Electrical,Electronics and Instrumentation Engineering. Vol. 03(11), pp. 13272-13284, Novembe r 2014.

[20] ManojKKowarandSourabhYadav,"BrainTumor Detction and Segmentation Using Histogram Thresholding."International Journal of Engineering and Advanced Technology. Vol. 01(04), pp. 16-20, April 2012.

[21] D.Selvaraj et.al / Indian Journal of Computer Science and Engineering (IJCSE)" Mri Brain Image Segmentation Techniques - A Review -ISSN : 0976-5166Vol. 4 No.5 Oct-Nov 2013