IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834, p- ISSN: 2278-8735.
PP 04-15
www.iosrjournals.org

MRI Brain Image Classification using GLCM Feature Extraction and Probabilistic Neural Networks

Priyanka Udayabhanu[1] ,Anjaly Viswan[2],Seema Padmarajan[3]
[231]*(Electronics and Communication Engineering ,SNGCE Kadayiruppu, Kerala,India)*

Abstract: *The project proposes an automatic support system for stage classification using artificial neural network (learning machine) and to detect Brain Tumor through k-means clustering methods for medical application. The detection of the Brain Tumor is a challenging problem, due to the structure of the Tumor cells. This project presents a segmentation method, k-means clustering algorithm, for segmenting Magnetic Resonance images to detect the Brain Tumor in its early stages and to analyze anatomical structures. The artificial neural network will be used to classify the stage of Brain Tumor that is benign, malignant or normal. Here discrete wavelet transform is used to analysis texture of an image. The segmentation results will be used as for early detection of Brain Tumor which will improves the chances of survival for the patient. Decision making was performed in two stages: feature extraction using GLCM and the classification using PNN-RBF network. The performance of this classifier was evaluated in terms of training performance and classification accuracies. The simulated results will be shown that classifier and segmentation algorithm provides better accuracy*

Keywords: *component, formatting, insert, style, styling (key words)*

I. Introduction

A brain cancer is a disease in which cells grow uncontrollably in the brain. Brain tumors have mainly two types, First is Benign tumors are unable of spreading beyond the brain itself. Benign tumors in the brain generally do not essential to be treated and their progress is self-limited. Sometimes they can cause complications because of their position and surgery or radiation can be helpful. And second is Malignant tumors are typically called brain cancer. These tumors can extent outside of the brain. Malignant tumors of the brain will always change into a problem if left untreated and aviolent approach is almost always warranted. Brain malignancies can be divided into two categories [11]: Primary brain cancer originates in the brain. Secondary or metastatic brain cancer extents to the brain from another site in the body Cancer arises when cells in the body (in this case brain cells) divide without control. Generally, cells divide in a structured manner. If cells keep separating uncontrollably when new cells are not needed, a mass of tissue forms, called a progress or tumor. The term cancer generally refers to malignant tumors, which can attack nearby tissues and can extent to other parts of the body. A benign tumor does not extent. An probabilistic neural network (PNN), generally called neural network (NN), is a mathematical model or computational model that is inspired by the structure and/or functional aspects of biological neural networks. A neural network contains of an interconnected group of probabilistic neurons(processing element), working in unison to solve specific problems. PNNs, like people, learn by example. The neuron has two modes of operations: The training/learning mode and the using/testing mode [16]**.** In mainly cases an PNN is an adaptive system that converts its structure based on external or internal information that flows through the network in the learning phase. Recent neural networks are non-linear statistical data modeling tools. They are generally used to model complex relationships between inputs and outputs or to find patterns in data**.** Decision making was performed in two stages: feature extraction using GLCM and the classification using PNN-RBF network. The performance of this classifier was evaluated in terms of training performance and classification accuracies. The simulated results will be shown that classifier and segmentation algorithm provides better accuracy.

II. System Analysis

Gray-level co-occurrence matrix (GLCM) is the statistical method of examining the textures that considers the spatial relationship of the pixels. The GLCM functions characterize the texture of an image by calculating how often pairs of pixel with specific values and in a specified spatial relationship that present in an image, forms GLCM. This forms the extraction of statistical measures from this matrix. The gray-co-matrix function in MATLAB creates a gray-level co occurrence matrix (GLCM) by calculating how often a pixel with the intensity (graylevel) value (for instance) row occurs in a specific spatial relationship to a pixel with the value by row and column. The relationship is defined as the picture elements with respect to the features present and the pixel to adjacent.

2.1 Image Segmentation

Segmentation problems are the bottleneck to achieve object extraction, object specific measurements, and fast object rendering from multi-dimensional image data. Simple segmentation techniques are based on local pixel-neighborhood classification. Such methods fail however to “see” global objects rather than local appearances and require often intensive operator assistance. The reason is that the “logic” of a object does not necessarily follow that of its local image representation. Local properties, such as textures, edgeness, and ridgeness etc. do not always represent connected features of a given object.

2.1.1 Region Growing Approach

Region growing technique segments image pixels that are belong to an object into regions. Segmentation is performed based on some predefined criteria. Two pixels can be grouped together if they have the same intensity characteristics or if they are close to each other. It is assumed that pixels that are closed to each other and have similar intensity values are likely to belong to the same object. The simplest form of the segmentation can be achieved through thresholding and component labeling. Another method is to find region boundaries using edge detection. Segmentation process, then, uses region boundary information to extract the regions. The main disadvantage of region growing approach is that it often requires a seed point as the starting point of the segmentation process. This requires user interaction. Due to the variations in image intensities and noise, region growing can result in holes and over segmentation. Thus, it sometimes requires post-processing of the segmentation result.

2.1.2 Clustering

Clustering can be considered the most important unsupervised learning problem, so it deals with finding a structure in a collection of unlabeled data. A cluster is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters

Clustering algorithms may be classified as listed below

- Exclusive Clustering
- Overlapping Clustering
- Hierarchical Clustering
- Probabilistic Clustering

In the first case data are grouped in an exclusive way, so that if a certain datum belongs to a definite cluster then it could not be included in another cluster. On the contrary the second type, the overlapping clustering, uses k-means sets to cluster data, so that each point may belong to two or more clusters with different degrees of membership. In this case, data will be associated to an appropriate membership value. A hierarchical clustering algorithm is based on the union between the two nearest clusters. The beginning condition is realized by setting every datum as a cluster. After a few iterations it reaches the final clusters wanted

2.1.3 K-means Segmentation

K-means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. Segmentation of brain tissues in nerve tissue, nerve tissue and neoplasm on medical pictures isn't solely of high interest in serial treatment observance of “disease burden” in medicine imaging, however conjointly gaining quality with the advance of image radio-controlled surgical approaches.When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as centers of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. Finally, this algorithm aims at minimizing an objective function, in this case a squared error function.

2.1.4 Hierarchical Segmentation

A hierarchical set of image segmentations is a set of several image segmentations of the same image at different levels of detail in which the segmentations at coarser levels of detail can be produced from simple merges of regions at finer levels of detail. A unique feature of hierarchical segmentation is that the segment or region boundaries are maintained at the full image spatial resolution for all segmentations. In a hierarchical segmentation, an object of interest may be represented by multiple image segments in finer levels of detail in the segmentation hierarchy, and may be merged into a surrounding region at coarser levels of detail in the segmentation hierarchy. If the segmentation hierarchy has sufficient resolution, the object of interest will be represented as a single region segment at some intermediate level of segmentation detail.

A goal of the subject analysis of the segmentation hierarchy is to identify the hierarchical level at which the object of interest is represented by a single region segment. The object may then be identified through its spectral and spatial characteristics. Additional clues for object identification may be obtained from the behavior of the image segmentations at the hierarchical segmentation level above and below the level at which the object of interest is represented by a single region.

2.1.5 Thresholding

The simplest method of image segmentation is called the thresholding method. This method is based on a clip-level (or a threshold value) to turn a gray-scale image into a binary image. The key of this method is to select the threshold value (or values when multiple-levels are selected). Several popular methods are used in industry including the maximum entropy method, Otsu's method (maximum variance), and k-means clustering. Recently, methods have been developed for thresholding computed tomography (CT) images. The key idea is that, unlike Otsu's method, the thresholds are derived from the radiographs instead of the (reconstructed) image.

Design Steps:
(1) Set the initial threshold T= (the maximum value of the image brightness + the minimum value of the image brightness)/2.
(2) Using T segment the image to get two sets of pixels B (all the pixel values are less than T) and N (all the pixel values are greater than T);
(3) Calculate the average value of B and N separately, mean ub and un.
(4) Calculate the new threshold: T= (ub+un)/2
(5) Repeat Second steps to fourth steps upto iterative conditions are met and get necessary region from the brain image.

2.1.6 Principal Component analysis

PCA is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called **principal components**. The number of principal components is less than or equal to the number of original variables. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to (i.e., uncorrelated with) the preceding components. Principal components are guaranteed to be independent only if the data set is jointly normally distributed. PCA is sensitive to the relative scaling of the original variables. Depending on the field of application, it is also named the discrete Karhunen–Loève transform (KLT), the Hotelling transform or proper orthogonal decomposition (POD).

Drawbacks

- Poor discriminatory power
- High computational load

III. System Design

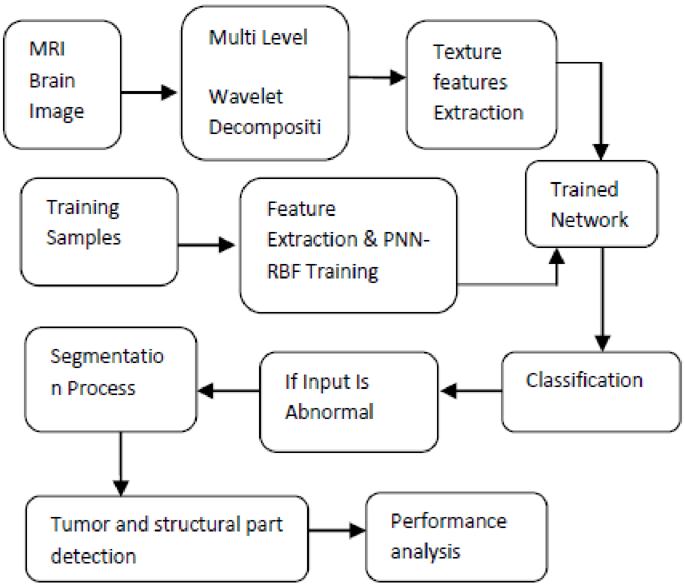

MRI Brain Image
Multi Level Wavelet Decompositi
Texture features Extraction
Training Samples
Feature Extraction & PNN-RBF Training
Trained Network
Segmentatio n Process
If Input Is Abnormal
Classification
Tumor and structural part detection
Performance analysis

Figure 1.block diagram of the system

3.1 DISCRETE WAVELET TRANSFORM

Over the past several years, the wavelet transform has gained widespread acceptance in signal processing in general and in image compression research in particular. In applications such as still image compression, discrete wavelets transform (DWT) based schemes have outperformed other coding schemes like the ones based on DCT. Since there is no need to divide the input image into non-overlapping 2-D blocks and its basis functions have variable length, wavelet-coding schemes at higher compression ratios avoid blocking artifacts. Because of their inherent multi -resolution nature, wavelet-coding schemes are especially suitable for applications where scalability and tolerable degradation are important. Recently the JPEG committee has released its new image coding standard, JPEG-2000, which has been based upon DWT. Basically we use Wavelet Transform (WT) to analyze non-stationary signals, i.e., signals whose frequency response varies in time, as Fourier Transform (FT) is not suitable for such signals.

3.1.1 Continuous wavelet transform

The 1-D continuous wavelet transform is given by:

$$W_f(a, b) = \int_{-\infty}^{\infty} x(t)\psi_{a,b}(t)dt \qquad \text{----- (3.3)}$$

$$x(t)= \frac{1}{C}\int_{0}^{\infty}\int_{-\infty}^{\infty} W_f(a,b)\psi_{a,b}(t)db\frac{da}{a^2} \quad \text{.... (3.4)} \qquad \text{Where } C = \int_{-\infty}^{\infty}\frac{|(\psi\omega)|^2}{\omega}d\omega<\infty$$

$\Psi(\omega)$is the Fourier transform of the mother wavelet Ψ(t). C is required to be finite, which leads to one of the required properties of a mother wavelet. Since C must be finite, then $\Psi(0) = 0$ to avoid a singularity in the integral, and thus the $\Psi(t)$ must have zero mean. This condition can be stated as $\int_{-\infty}^{\infty}\psi(t)dt$ = 0 and known as the admissibility condition.

3.1.1 1-D Discrete wavelet transform

The discrete wavelets transform (DWT), which transforms a discrete time signal to a discrete wavelet representation. The first step is to discretize the wavelet parameters, which reduce the previously continuous basis set of wavelets to a discrete and orthogonal / orthonormal set of basis wavelets.

$\psi m,n(t) = 2^{m/2}\ \psi(2^{m}t - n)$; m, n $\in$Zsuch that - $\infty<$ m, n $<\infty$ ------- (3.5)

The 1-D DWT is given as the inner product of the signal x(t) being transformed with each of the discrete basis functions. Wm,n = < x(t), ψm,n(t) > ; m, n $\in$ Z ----(3.6)
The 1-D inverse DWT is given as: x (t) = $\sum\sum W_{m,n}\psi_{m,n}(t)$; m, n $\in$ Z---- (3.7)
m n

3.1.3 2-D wavelet transform

The 1-D DWT can be extended to 2-D transform using separable wavelet filters. With separable filters, applying a 1-D transform to all the rows of the input and then repeating on all of the columns can compute the 2-D transform. When one-level 2-D DWT is applied to an image, four transform coefficient sets are created. As depicted in Figure 2(c), the four sets are LL, HL, LH, and HH, where the first letter corresponds to applying either a low pass or high pass filter to the rows, and the second letter refers to the filter applied to the columns.

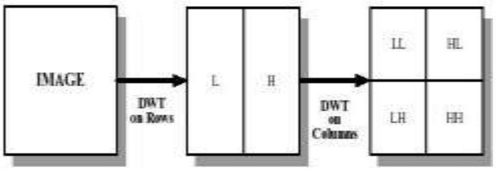


Figure 2 Block Diagram of DWT

(a) Original Image (b) Output image after the 1-D applied on Row input (c) Output image after the second 1-D applied on row input

The Two-Dimensional DWT (2D-DWT) converts images from spatial domain to frequency domain. At each level of the wavelet decomposition, each column of an image is first transformed using a 1D vertical analysis filter-bank. The same filter-bank is then applied horizontally to each row of the filtered and sub sampled data. One-level of wavelet decomposition produces four filtered and sub sampled images, referred to as sub bands. The upper and lower areas of Fig. 2(b), respectively, represent the low pass and high pass coefficients after vertical 1D-DWT and sub sampling. The result of the horizontal 1D-DWT and sub sampling to form a 2D-DWT output image is shown in Fig.2(c).

We can use multiple levels of wavelet transforms to concentrate data energy in the lowest sampled bands. Specifically, the LL sub band in fig 2.1(c) can be transformed again to form LL2, HL2, LH2, and HH2 sub bands, producing a two-level wavelet transform. An (R-1) level wavelet decomposition is associated with R resolution levels numbered from 0 to (R-1), with 0 and (R-1) corresponding to the coarsest and finest resolutions. The straight forward convolution implementation of 1D-DWT requires a large amount of memory and large computation complexity. An alternative implementation of the 1D-DWT, known as the lifting scheme, provides significant reduction in the memory and the computation complexity. Lifting also allows in-place computation of the wavelet coefficients. Nevertheless, the lifting approach computes the same coefficients as the direct filter-bank convolution.

3.1.4 2-D Transform Hierarchy

The 1-D wavelet transform can be extended to a two-dimensional (2-D) wavelet transform using separable wavelet filters. With separable filters the 2-D transform can be computed by applying a 1-D transform to all the rows of the input, and then repeating on all of the columns.

LL1	HL1
LH1	HH1

Fig.3. Sub band Labeling Scheme for a one level, 2-D Wavelet Transform

3.1.5 Wavelet Computation

In order to obtain an efficient wavelet computation, it is important to eliminate as many unnecessary computations as possible. A careful examination of the forward and reverse transforms shows that about half the operations either lead to data which are destroyed or are null operations (as in multiplication by 0) The one-dimensional wavelet transform is computed by separately applying two analysis filters at alternating even and odd locations. The inverse process first doubles the length of each signal by inserting zeros in every other position, then applies the appropriate synthesis filter to each signal and adds the filtered signals to get the final reverse transform.

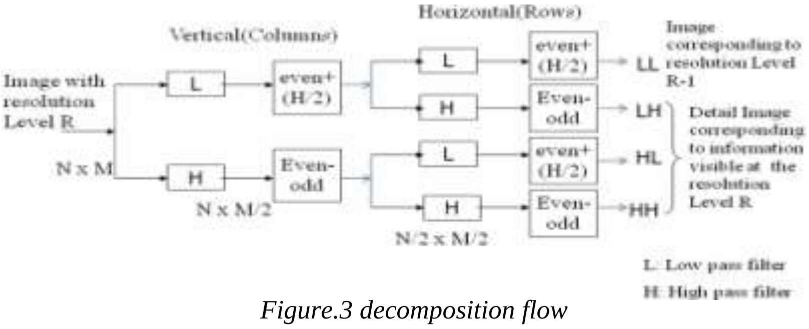


Figure.3 decomposition flow

3.1.6 Forward transform

Step1: Column wise processing to get H and L : H = (Co-Ce)….. (1) L = (Ce- _*H*/2_)…… (2)

Where Co and Ce is the odd column and even column wise pixel values

Step 2: Row wise processing to get LL,LH,HL and HH, Separate odd and even rows of H and L,

Hodd – odd row of H,Lodd- odd row of L,Heven- even row of H,Leven - even row of L,LH = Lodd-

Leven LL = Leven – (*LH* / 2) ,HL = Hodd – Heven ,HH = Heven – (*HL* / 2)

3.2 TEXTURE ANALYSIS

Texture is that innate property of all surfaces that describes visual patterns, each having properties of homogeneity. It contains important information about the structural arrangement of the surface, such as; clouds, leaves, bricks, fabric, etc. It also describes the relationship of the surface to the surrounding environment. In short, it is a feature that describes the distinctive physical composition of a surface. Texture properties include Coarseness,Contrast,Directionality,Line-likeness,Regularity,Roughness.Texture is one of the most important defining features of an image. It is characterized by the spatial distribution of gray levels in a neighborhood [8]. In order to capture the spatial dependence of gray-level values, which contribute to the perception of texture, a two-dimensional dependence texture analysis matrix is taken into consideration. This two-dimensional matrix is obtained by decoding the image file; jpeg, bmp, etc.

Methods of Representation:

There are three principal approaches used to describe texture; statistical, structural and spectral…

- Statistical techniques characterize textures using the statistical properties of the grey levels of the points/pixels comprising a surface image. Typically, these properties are computed using: the grey level co-occurrence matrix of the surface, or the wavelet transformation of the surface.

- Structural techniques characterize textures as being composed of simple primitive structures called “texels” (or texture elements). These are arranged regularly on a surface according to some surface arrangement rules.
- Spectral techniques are based on properties of the Fourier spectrum and describe global periodicity of the grey levels of a surface by identifying high-energy peaks in the Fourier spectrum [9].

For optimum classification purposes, what concern us are the statistical techniques of characterization… This is because it is these techniques that result in computing texture properties… The most popular statistical representations of texture are:Co-occurrence Matrix, Tamura Texture, Wavelet Transform
Originally proposed by R.M. Haralick, the co-occurrence matrix representation of texture features explores the grey level spatial dependence of texture [2]. A mathematical definition of the co-occurrence matrix is as follows [4]:

- Given a position operator ***P(i,j)***,
- let ***A*** be an n x n matrix
- whose element ***A[i][j]*** is the number of times that points with grey level (intensity) ***g[i]*** occur, in the position specified by ***P***, relative to points with grey level ***g[j]***.

- Let ***C*** be the ***n* x *n*** matrix that is produced by dividing ***A*** with the total number of point pairs that satisfy ***P***. ***C[i][j]*** is a measure of the joint probability that a pair of points satisfying ***P*** will have values ***g[i]***, ***g[j]***.

- ***C*** is called a co-occurrence matrix defined by ***P***.

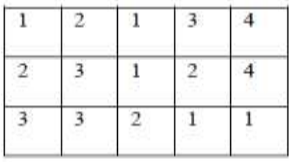
Examples for the operator ***P*** are: “***i*** above ***j***”, or “***i*** one position to the right and two below ***j***”, etc.

This can also be illustrated as follows… Let *t* be a translation, then a co-occurrence matrix *Ct*of a region is defined for every grey-level *(a, b)* by [1]:

$$C_t\,(a,b) = card\{(s,s+t) \in R^2 \mid A[s] = a, A[s+t] = b\}$$

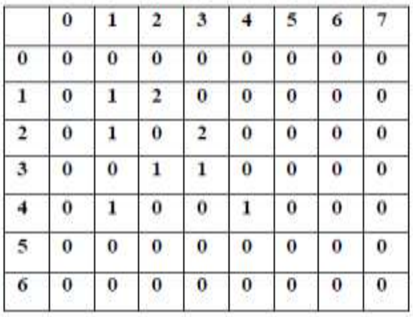
Here, *Ct(a, b)* is the number of site-couples, denoted by *(s, s + t)* that are separated by a translation vector *t*, with *a* being the grey-level of *s*, and *b* being the grey-level of *s + t*.

For example; with an 8 grey-level image representation and a vector *t* that considers only one neighbor, we would find [1]:

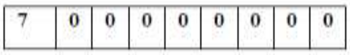


1	2	1	3	4
2	3	1	2	4
3	3	2	1	1

Figure.4. Image example

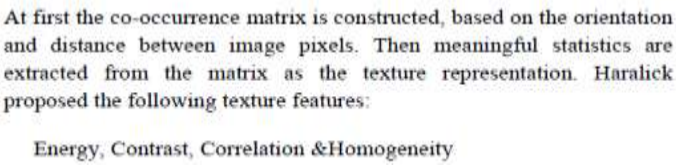


	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	0	0	0	0	0
2	0	1	0	2	0	0	0	0
3	0	0	1	1	0	0	0	0
4	0	1	0	0	1	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0



7	0	0	0	0	0	0	0	0

Figure.5. Classical Co-occurrence matrix



At first the co-occurrence matrix is constructed, based on the orientation and distance between image pixels. Then meaningful statistics are extracted from the matrix as the texture representation. Haralick proposed the following texture features:

Energy, Contrast, Correlation &Homogeneity

Hence, for each Haralick texture feature, we obtain a co-occurrence matrix. These co-occurrence matrices represent the spatial distribution and the dependence of the grey levels within a local area. Each ***(i,j)*** th entry in the matrices, represents the probability of going from one pixel with a grey level of '***i***' to another with a grey level of '***j***' under a predefined distance and angle. From these matrices, sets of statistical measures are computed, called feature vectors.

1. Energy**:** It is a gray-scale image texture measure of homogeneity changing, reflecting the distribution of image gray-scale uniformity of weight and texture..

E=∑ ∑ P(X,Y) p(x,y) ^2 is the GLC M

2. Contrast**:** Contrast is the main diagonal near the moment of inertia, which measure the value of the matrix is distributed and images of local changes in number, reflecting the image clarity and texture of shadow depth.
Contrast I=∑∑(x-y) ^2 p(x,y)

3. Correlation Coefficient**:** Measures the joint probability occurrence of the specified pixel pairs.
Correlation**:** sum(sum((x- μx)(y-μy)p(x , y)/σxσy))

4. Homogeneity**:** Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.
Homogeneity = sum(sum(p(x , y)/(1 + [x-y])))

3.3 PROBABILISTIC NEURAL NETWORKS

Probabilistic (PNN) and General Regression Neural Networks (GRNN) have similar architectures, but there is a fundamental difference: Probabilistic networks perform classification where the target variable is categorical, whereas general regression neural networks perform regression where the target variable is continuous. If you select a PNN/GRNN network, DTREG will automatically select the correct type of network based on the type of target variable.

Fortunately, you won't have to write such lines of code. All of the details of designing this network are built into design functions newrbe and newrb, and their outputs can be obtained with sim.

A PNN is an implementation of a statistical algorithm called kernel discriminant analysis in which the

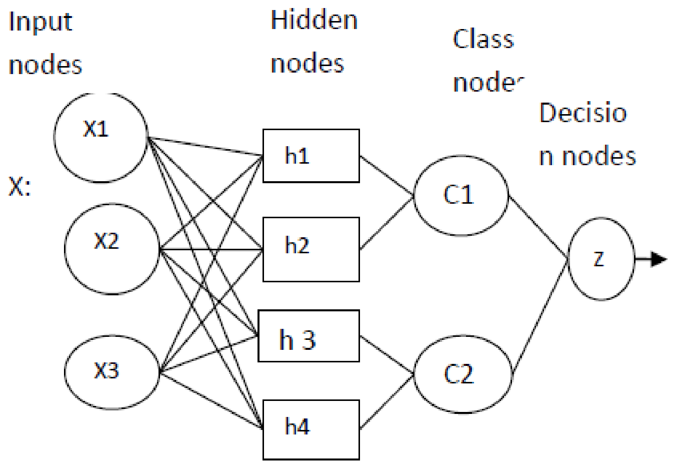


Figure.6.Architecture of a PNN

All PNN networks have four layers:
1. Input layer — There is one neuron in the input layer for each predictor variable. In the case of categorical variables, *N*-1 neurons are used where *N* is the number of categories. The input neuron (or processing before the input layer) standardizes the range of the values by subtracting the median and dividing by

the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.

2. Hidden layer — This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along with the target value. When presented with the *x* vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the RBF kernel function using the sigma value(s).

The resulting value is passed to the neurons in the pattern layer.

3. Pattern layer / Summation layer — The next layer in the network is different for PNN networks and for GRNN networks. For PNN networks there is one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron; the weighted value coming out of a hidden neuron is fed only to the pattern neuron that corresponds to the hidden neuron's category. The pattern neurons add the values for the class they represent (hence, it is a weighted vote for that category).

4. Decision layer — The decision layer is different for PNN and GRNN networks. For PNN networks, the decision layer compares the weighted votes for each target category accumulated in the pattern layer and uses the largest vote to predict the target category.

How PNN network work**:**

Although the implementation is very different, probabilistic neural networks are conceptually similar to *K-Nearest Neighbor* (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. Consider this figure.7

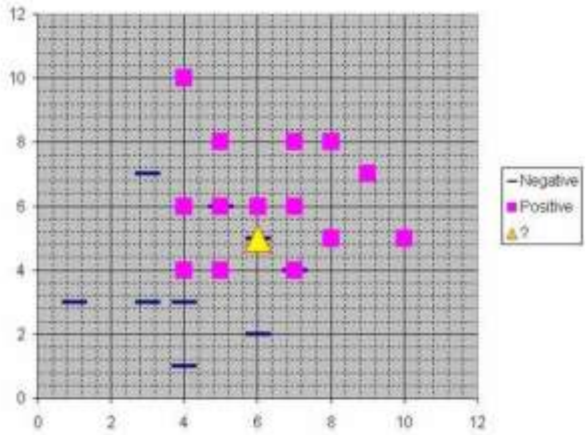


Figure.7

Assume that each case in the training set has two predictor variables, *x* and *y*. The cases are plotted using their *x,y* coordinates as shown in the figure. Also assume that the target variable has two categories, *positive* which is denoted by a square and *negative* which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values *x*=6, *y*=5.1. Should we predict the target as positive or negative?

Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

The nearest neighbor classification performed for this example depends on how many neighboring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

A probabilistic neural network builds on this foundation and generalizes it to consider all of the other points. The distance is computed from the point being evaluated to each of the other points, and a *radial basis function* (RBF) (also called a *kernel function*) is applied to the distance to compute the weight (influence) for each point.

The radial basis function is so named because the radius distance is the argument to the function.

Weight = RBF (*distance*)
The further some other point is from the new point, the less influence it has.

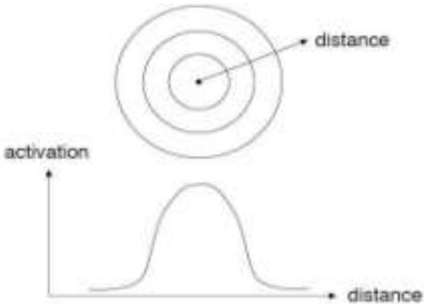


Figure.8.Radial Basis Function

Different types of radial basis functions could be used, but the most common is the Gaussian function:

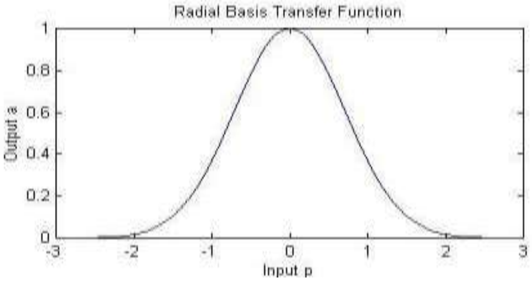


Figure.9. Radial Bias Transfer Function

Removing unnecessary neurons

One of the disadvantages of PNN models compared to multilayer perceptron networks is that PNN models are large due to the fact that there is one neuron for each training row. This causes the model to run slower than multilayer perceptron networks when using scoring to predict values for new rows. DTREG provides an option to cause it remove unnecessary neurons from the model after the model has been constructed. Removing unnecessary neurons has three benefits:

1. The size of the stored model is reduced.
2. The time required to apply the model during scoring is reduced.
3. Removing neurons often improves the accuracy of the model.

The process of removing unnecessary neurons is an iterative process. Leave-one-out validation is used to measure the error of the model with each neuron removed. The neuron that causes the least increase in error (or possibly the largest reduction in error) is then removed from the model. The process is repeated with the remaining neurons until the stopping criterion is reached. When unnecessary neurons are removed, the “Model Size” section of the analysis report shows how the error changes with different numbers of neurons. You can see a graphical chart of this by clicking Chart/Model size. There are three criteria that can be selected to guide the removal of neurons:

1. Minimize error – If this option is selected, then DTREG removes neurons as long as the leave-one-out error remains constant or decreases. It stops when it finds a neuron whose removal would cause the error to increase above the minimum found.
2. Minimize neurons – If this option is selected, DTREG removes neurons until the leave-one-out error would exceed the error for the model with all neurons.
3. # of neurons – If this option is selected, DTREG reduces the least significant neurons until only the specified number of neurons remain.

IV. Results And Discussion

4.1 *Performance Analysis*

The performance of classifier can be evaluated through following parameters,
1. Sensitivity**:** It measures the proportion of actual positives which are correctly identified

Sensitivity = Tp. / (Tp + Fn), Where, Tp = True Positive: Defective correctly classified as defective metal Fn = False negative: Defective incorrectly classified as normal

2. Specificity**:** It measures the proportion of negatives which are correctly identified.
Specificity = Tn./(Fp + Tn), Where,Fp = False Positive: Normal incorrectly classified as defective one

Tn = True negative: Normal correctly classified as normal

3. Total accuracy**:** (Tp+Tn)./(Tp+Tn+Fp+Fn)

The network generates the performance metrics,

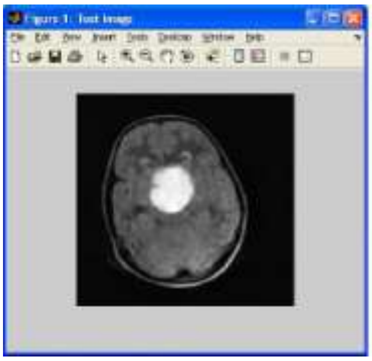
- Sensitivity: 85.7143%,
- Specificity: 100%,
- Accuracy: 90.9091%

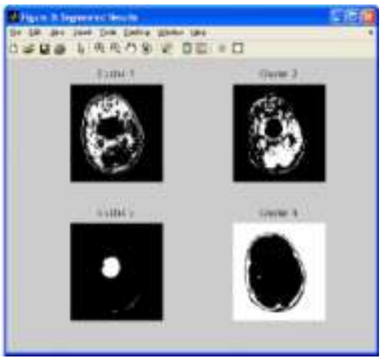
1. Tumor Area Measurement

The segmented abnormal region area will be determined by counting the number of nonzero pixels and one pixel's area.The one point area is 0.264 mm. The tumor area is, Area = sqrt(Pixel Count) * 0.264 mm. ^2

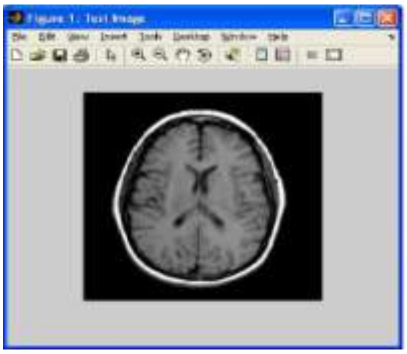
4.2 Results

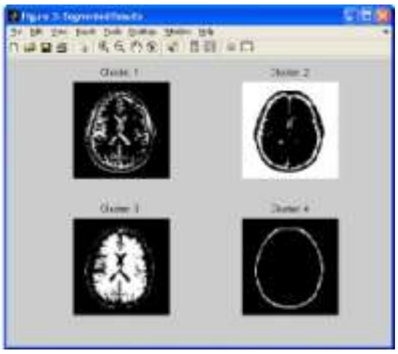
1. *MRI BRAIN IMAGE WITH BENIGN CASE*



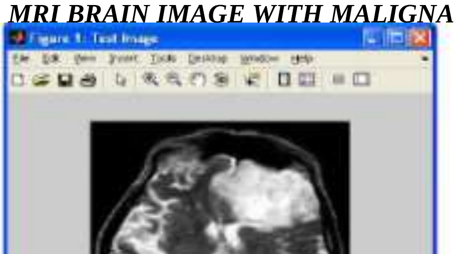


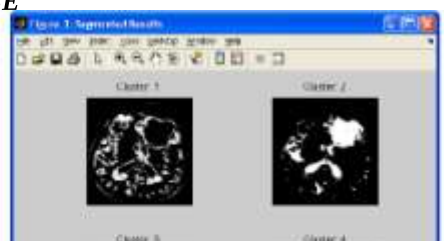
2. *MRI BRAIN IMAGE WITH NORMALCASE*





3. *MRI BRAIN IMAGE WITH MALIGNANT CASE*





V. Conclusion

The project presented that automated brain image classification for early stage abnormality detection with use of neural network classifier and spotting of tumor was done with image segmentation. Pattern recognition was performed using probabilistic neural network with radial basis function and pattern will be characterized with the help of fast discrete curvelet transform and haralick features analysis. Here. Spatial k-means clustering algorithm was utilized effectively for accurate tumor detection to measure the area of abnormal region. From an experiment, system proved that it provides better classification accuracy with various stages of test samples and it consumed less time for process.

References

[1]. N. Kwak, and C. H. Choi, “Input Feature Selection for Classification Problems”, IEEE Transactions on Neural Networks, 13(1), 143–159, 2002.

[2]. E. D. Ubeyli and I. Guler, “Feature Extraction from Doppler Ultrasound Signals for Automated Diagnostic Systems”, Computers in Biology and Medicine, 35(9), 735–764, 2005.

[3]. D.F. Specht, “Probabilistic Neural Networks for Classification, mapping, or associative memory”, Proceedings of IEEE International Conference on Neural Networks, Vol.1, IEEE Press, New York, pp. 525-532, June 1988.

[4]. D.F. Specht, “Probabilistic Neural Networks” Neural Networks, vol. 3, No.1, pp. 109-118, 1990.

[5]. Georgiadis. Et all , “ Improving brain tumor characterization on MRI by probabilistic neural networks and non-linear transformation of textural features”,Computer Methods and program in biomedicine, vol 89, pp24-32, 2008

[6]. Kaus M., “Automated segmentation of MRI brain tumors”, Journal of Radiology,vol. 218, pp. 585-591,2001

[7]. Kornel, P., Bela, M., Rainer, S., Zalan, D., Zsolt, T. and Janos, F.,“Application of neural network in medicine”, Diag. Med. Tech.,vol. 4,issue 3,pp: 538-54 ,1998.

[8]. Mammadagha Mammadov, Engintas ,”An improved version of back propagation algorithm with effective dynamic learning rate and momentum “ ,Inter.Conference on Applied Mathematics ,pp:356-361, 2006.

[9]. Messen W, Wehrens R, Buydens L, “Supervised Kohonen networks for classification problems“,Chemometrics and Intelligent Laboratory Systems, vol.83,pp:99-113,2006.

[10]. Orr M.J.L., Hallam J., Murray A., and Leonard .T, “Assessing rbf networks using delve," International Journal of Neural Systems, vol. 10, issue 5, pp. 397-415, 2000