A Comparative Study of Different Feature Extraction and Classification Methods for Recognition of Handwritten Kannada Numerals

Mamatha Hosalli Ramappa[1] and Srikantamurthy Krishnamurthy[2]

[1]Department of Information Science and Engineering
P E S Institute of Technology
(West Campus) Bangalore, India
[2]Department of Computer Science and Engineering
P E S Institute of Technology
(South Campus) Bangalore, India

Abstract

The feature extraction and classification method(s) used to recognize handwritten characters play an important role in Handwritten Character Recognition applications. A suitable feature extractor and a good classifier play a very important role in achieving high recognition rate for a recognition system. If we want to develop a new feature extractor for a script, it will help us if we have the knowledge of the recognition ability of the existing feature extractor. Kannada is a major south Indian script spoken by about 50 million people. This paper examines a variety of feature extraction approaches and classification methods which have been used in various Optical Character Recognition applications which are designed to recognize handwritten numerals of Kannada script. The study has been conducted using 8 different features computed from zonal extraction, image fusion, radon transform, fan beam projections, directional chain code, discrete fourier transform, run length count and curvelet transform along with ten different classifiers like Euclidean distance, Chebyshev distance, Manhattan (city block) distance, Cosine distance, K-NN, K-means, K-medoids, Linear classifier, Artificial Immune system and Classifier fusion are considered.

Keywords: *Comparative study, classifier, feature extraction, Recognition of Handwritten Kannada Numerals*

1. Introduction

A character is a system consisting of symbols that are used to represent information of a particular language. A character is represented as a physical entity using a number of writing tools so that the information present can be transmitted to the reader. Character recognition is a topic being studied in depth. Human's ability to recognize characters is being studied in its advanced form and systems are being developed to simulate this ability. Character recognition plays an important role in automating insertion and human-computer interface.

The Handwritten numeral recognition is an important area of research for its applications in post office, banks and other organizations. An extensive attention has been received by handwritten character recognition (HCR) in the fields of academics and production. Recognition systems can be classified as on-line or off-line. The process of finding letters and words present in digital image of handwritten text is called as off-line handwriting recognition. Several methods of recognition of English, Latin, Arabic, Chinese scripts have been excellently reviewed in [1, 2, 3, 4]. Research in HCR has become popular because of many practical applications such as reading aid for the blind, recognition of numbers on bank cheques, automatic pin code reading for sorting of postal mail *etc*.

A lot of work has been done on the recognition of printed characters of Indian languages. On the other hand, attempts made on the recognition of handwritten characters are few. Most of the research in this area is concentrated on recognition of off-line handwritten characters for the scripts like Devanagari and Bangla. From the literature survey it can be seen that there is a lot of demand for character recognition systems for Indian scripts and an excellent review has been done on the Optical Character Recognition (OCR) for Indian languages [5]. A detailed study and analysis of optical character recognition research on south Indian Scripts can be seen in [6].

An efficient and novel method for recognition of machine printed and handwritten Kannada numerals using Crack codes and Fourier descriptors is reported in [7]. Similar shape character recognition for different scripts like Arabic/Persian, Devnagari, English, Bangla, Oriya, Tamil, Kannada, Telugu *etc.*, using F-ratio Based Weighted Feature Extraction is presented in [8]. Rajput and Mali [9] have proposed an efficient method for recognition of isolated Devanagari handwritten numerals based on Fourier descriptors. In [10] zone centroid is computed and the image is further divided in to n equal zones. Average distance from the zone centroid to the each pixel present in the zone is computed. This procedure is repeated for all the zones present in the numeral image. Finally ‘n’ such features are extracted for classification and recognition.

Selection of a suitable feature extraction method plays an important role in achieving high recognition performance. A survey of feature extraction methods for character recognition is reported in [11]. Literature survey reveals that the automatic recognition of handwritten numeral has been the subject of intensive research during the last few decades. Numeral identification is very vital in applications such as interpretation of identity numbers, vehicle registration numbers, pin codes, etc. In the Indian context, it is evident that handwritten numeral recognition is still a fascinating area of research and efforts are being made to design a robust optical character recognition (OCR), in particular for handwritten Kannada numeral recognition.

To get an idea of the different techniques used in recognition and to provide a new benchmark for future research, results of different feature extraction methods with different classifiers and a comparative study of Kannada handwritten numeral recognition results are reported in this paper. To compare the performance of recognition systems, eight different features computed from zonal extraction, image fusion, radon transform, fan beam projections, directional chain code, discrete fourier transform, run length count and curvelet transform are used. Also ten different classifiers like Euclidean distance, chebyshev distance, Manhattan (city block) distance, Cosine distance, k-NN ,K-means, K-medoids, Linear classifier, Artificial Immune system and Classifier fusion are considered.

A pattern recognition model when used as a feature extractor in handwritten recognition requires training the machines (systems) with handwritten samples of different persons. In order to train a machine, a large number of training samples are required. If the number of samples is small, the machine may not learn adequately. There is a lot of impact of size of training data set on the recognition performance of a classifier. Training a classifier with adequate number of handwritten samples enrich it with generalization ability. It is essential to know the effect of size of training dataset on the recognition performance of a feature extraction method. Here an attempt is made to measure the performance of the classifier by training the classifier with two training data sets.

In practical classification applications, if the number of classes and multiple feature sets for pattern samples are given, a desirable recognition performance can be achieved by fusing the decisions. This technique is generally known as distributed data fusion or decision fusion where individual decisions are first made based on different feature sets and then they are reconciled or combined into a global decision. Here, we have made an attempt to fuse the decisions of the classifiers.

The rest of the paper is organized as follows. Section 2 describes the Kannada script, Section 3 discusses the data set and preprocessing, Sections 4 and 5 briefly discuss the feature extraction and classification methods respectively. Section 6 gives a study on some significant factors. Finally in Sections 7 and 8, comparative study and conclusions are made.

2. Description of the Kannada Script

Kannada also called as Canarese, is the official language of the state of Karnataka which is present in the southern part of India. Described as 'sirigannada', it is one of the earliest languages evidenced epigraphically in India. The language is spoken by about 50 million people spread over the states of Karnataka, Tamil Nadu, Andhra Pradesh and Maharashtra. The visual form of the Kannada language is the Kannada script. The Kannada script originated from the southern Bramhi Lipi during the period of Ashoka. It underwent a lot of changes from time to time during the reign of Sathavahanas, Kadambas, Gangas, Rastrakutas and Hoysalas [12]. The modern Kannada script emerged in the thirteenth century. It is also used to write Tulu, Konkani and Kodava languages.

The Kannada script has a large number of structural features which are common with other Indian language scripts. Kannada has 49 basic characters which are classified into three categories: swaras(vowels), vyanjans(consonants) and yogavaahas (part vowel, part consonants). The script also has 10 Kannada numerals. A sample sheet of Kannada handwritten numerals is shown in Figure 1.

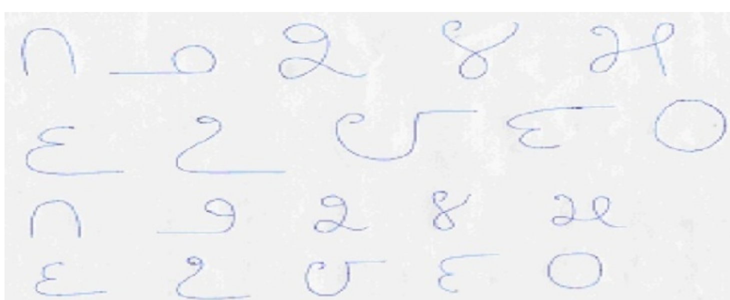


Figure 1. A Sample Sheet of Kannada Handwritten Numerals 0 to 9

The challenging part of Kannada handwritten character recognition is the distinction between the similar shaped components. The style of writing characters is highly different and they come in various sizes and shapes. Same numeral may take different shapes and conversely two or more different numerals of a script may take similar shape. A very small variation between two characters or numerals leads to recognition complexity and reduces the accuracy rate of the recognition system. Some examples of the similar shaped numerals are as shown in Figure 2

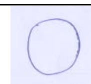
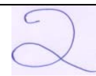
Numeral 0 and 1		
Numeral 6 and 9		
Numeral 3 and 7		

Figure 2. Examples of some Similar Shaped Numerals

3. Data Set and Preprocessing

For OCR performance evaluation a standard database of character images is required which is lacking in Kannada script. In order to build the dataset for our experimentation, a dataset of the 10 numerals was created by collecting various handwritten samples from writers belonging to different categories comprising of different age group like students, officials, housewives etc. The skew in the documents has not been considered. A sample image of scanned document is shown in Figure 1. The samples were scanned through a flatbed HP scanner at 300 dpi. Isolated characters were obtained by manual cropping. Thus 100 different samples of each numeral were created with the total of 1000 samples.

Initially the color images were converted to gray scale and in turn the gray scale images were converted to binary using global threshold method. Thinning is applied on the binary image. Thinning is an image preprocessing operation performed to make the image crisper by reducing the binary-valued image regions to lines that approximate the skeletons of the region. Region labeling is performed on the thinned binary image of the numeral and a minimum rectangle bounding box is inserted over the numeral. The bounding box image would be of variable size due to different style and size of numeral. Hence this image is resized to desired size.

4. Feature Extraction

4.1. Zonal based Feature Extraction

The preprocessed image is resized to 60 x 60.The resized image is divided into zones or blocks of 5 x 5 to obtain the features. A feature vector is then computed by considering the number of on pixels in each zone. For each zone if the number of on pixels is greater than 5% of total pixels, then the value one is stored for that block. The size of the feature vector is 144.The extracted features are then used for training and classification [13].

4.2. Image Fusion

The image fusion technique has been described in [14]. Here the extracted features of the several images corresponding to each handwritten numeral are fused to generate patterns, which are stored in 8x8 matrices, irrespective of the size of images. Zonal based feature extraction algorithm has been used to extract the features of handwritten Kannada numerals.

Feature Extraction and Fusion

The feature extraction module includes the following steps:

- Division of the normalized image into 64 zones of equal size.
- Creation of a Pattern Matrix (Mp_{tj}) of size 8x8.
- For each zone if the number of on pixels is greater than 5% of total pixels, store 1 in Mp_{tj}.
- Fusion of the reduced image Mp_{tj} with Pm_j using the equation (1)

$$P_{M_j}^{New} = \left(\frac{1}{num+2}\right) * \left(num * P_{M_j}^{old} + M_{P_{tj}}\right)\ 0 \le j \le 9 \quad (1)$$

Here, P_{Mj}^{New} is the fused pattern matrix obtained after fusing training images contained in Mptj and P_{Mj}^{Old} (already stored in the pattern matrix table). P_{Mj}^{New} is copied back to the table along with num increased by 1. The content of each cell of fused pattern matrix represents the probability of occurrence of a white pixel that is mapped with the test image to a typical numeral [15].

4.3. Radon Transform

Radon transform is used as one of the feature extraction methods [16].In Radon transform, 50 diverging beams are used to compute the features. It is seen from the accumulator data that the projections taken from 0 to 180 degree are exactly equal to the projections taken from 181 to 360 degree. Average value of the obtained projection data is taken to build the feature vector. Average was taken for 180 degree rotation of angle theta that is once we get the projection vector, we take the average of the pixels at the co-ordinates from 0 degree to 179 degrees. Hence size of feature vector for one numeral is 703 x 1 [17].

4.4. Fan Beam Projection

Fan beam projection is a variation [18] of Radon transform. The fan-beam function computes projections of an image matrix along specified directions except that the projections are taken in a different way from that of Radon transform.

Features were computed using fan-beam geometry [17]. For Fan-beam, 55 diverging beams are taken. The first step is to determine the distance D from the fan-beam source to the center of rotation (Figure 3). D must be large enough to ensure that the fan-beam source is outside of the image at all rotation angles. D is taken a few pixels larger than half the diagonal image distance, where the diagonal image distance is, d=$\sqrt{i^2 - j^2}$,where i and j are rows and columns of the image respectively. Fan-beam takes projections at different angles by rotating the source around the center pixel at θ degree intervals. These projection data is considered as feature vector.

It can be seen from the accumulator data of Fan beam that after 180 degree the signal repeats itself in the reverse direction. This is because projections taken from 0 to 180 degree are exactly equal to the projections taken from 181 to 360 degree. The average value of the obtained projection data is computed in order to build the feature vector. For Fan-beam the average of the projections of one direction which is the average of 55 parallel projections was computed. Hence the size of feature vector for one numeral is 1×360[17].

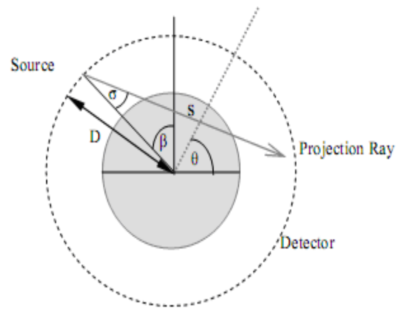


Figure 3. Fan-beam Geometry

4.5. Directional Features

Histograms of direction chain code of the contour points of the handwritten Kannada numerals are used as features for recognition [19]. Here the image is resized to 30x30 and is divided into blocks of 10x10 each. Thus, we have nine zones with each zone having eight features. In total we have 72 features of each numeral [20].

4.6. Discrete Fourier Transform

The literature survey reports that the most of the work for recognition of numeral addresses feature extraction methods in spatial domain and relatively few works exist for feature extraction in frequency domain [7]. When working with digital images, a finite number of discrete samples have been considered most of the time. These samples are the pixels that compose an image. In order to access the geometric characteristics of an image in the spatial domain, the Fourier Transform will be used because the Fourier domain decomposes the image into its sinusoidal components, which makes it easy to process certain frequencies of the image. The Discrete Fourier Transform (DFT) is the sampled Fourier Transform, which is large enough to fully describe the image in spatial domain rather than all the frequencies forming an image.

The formula to compute the Discrete Fourier transform is Equation (2).

$$X(k) = \sum_{j=1}^{N} x(j)\omega_N^{(j-1)(k-1)} \qquad (2)$$

where $\omega_N = e^{(-2\pi i)/N}$ and N is the total number of samples.

We obtain the average pixel values of rows and columns for each block and then extract the features by applying the Discrete Fourier Transform for each of the row and column vectors [21].

4.7. Run Length Count

From the literature survey we can observe that most of the character recognition methods focus on extracting either statistical feature such as zoning, moments *etc.*, or structural features based on the geometry of the character. In this paper we have proposed a method, which attempts to combine both the statistical and structural features. In this method, we divide the entire image into 9 equal sized zones. In a binary image, whenever a pixel value changes from 0 to 1 or 1 to 0 it indicates that the information represents an edge. This information is very significant as it denotes the geometry of the character and helps in identifying the character [22]. In order to capture this information, we have used Run Length Count (RLC) technique. In this method, for every zone, we find the Run Length count in horizontal and vertical direction. A total of 18 features will be extracted for each character and this will serve as feature vector.

4.8. The Curvelet Transform

The next feature extraction technique that we have used to extract the features of a numeral is the Curvelet transform. We have used the Curvelet Transform because it extracts features efficiently from images which contain a large number of C^2 curves (*i.e.*, an image which has a large number of long edges) [23].

Wrapping based discrete Curvelet transform using Curvelab-2.1.2 is applied to find the coefficients and to create feature vectors for every 256*256 image in the database. In this experiment we have used the default orientation and 5 levels of discrete Curvelet decomposition. Hence for an image of size 256*256, Curvelet coefficients in five different scales are obtained. Then we have applied standard deviation as a dimensionality reduction technique in order to reduce the number of features. It is seen that by applying standard deviation we can reduce 29,241 features obtained in scale 1 to 171, 94,777 features obtained in scale 2 to 426, 1,64,953 features obtained in scale 3 to 348, 1,80,601 features obtained in scale 4 to 322 and 1,84,985 features obtained in scale 5 to 316[24].

5. Classification

5.1. Euclidean Distance Metric

Euclidean distance [25] between P and Q is computed using the equation (3) given below.

$$D = \|P - Q\| \qquad (3)$$

Where P represents the input test image and Q is the trained images of the classes in the database. The nearest neighbor technique assigns the test numeral to a class that has the minimum distance. The corresponding numeral is declared as recognized numeral [15].

5.2. Chebyshev Distance Metric

The Chebyshev distance[25] between two vectors or points p and q, with standard coordinates p_i and q_i, respectively is given by the equation (4)[15],

$$D(p,q) \coloneqq max_i(|p_i - q_i|) \quad 0 \le i \le n \qquad (4)$$

5.3. Manhattan Distance Metric

The Manhattan distance [25] function computes the distance that would be travelled to get from one data point to the other if a grid-like path is followed. The Manhattan distance between two items is the sum of the differences of their corresponding components. The formula for this distance between a point P= (p_1, p_2 ...) and a point Q=

(q_1, q_2 …), with standard coordinates p_i and q_i respectively is given by the following equation (5) [15].

$$D(x) = \sum_{i=1}^{n} |p_i - q_i| \qquad (5)$$

5.4. Cosine Distance Metric

Cosine Distance [25] is a type of Pearson measure, which considers the relative differences (*e.g.*, A*B/|A|.|B|) assuming that the scale is uniform (that the distance from zero is relative). In some cases this can give better results, particularly where the data is not 'normally' distributed. It is a measure of similarity between two vectors by measuring the cosine of the angle between them. The result of the Cosine function is equal to 1 when the angle is 0, and it is less than 1 when the angle is of any other value. Calculating the cosine of the angle between two vectors thus determines whether two vectors are pointing in roughly the same direction. Given two vectors of attributes, P and Q, the cosine similarity, θ, is represented using a dot product and magnitude as shown in equation (6)[15].

$$similarity = cos(\theta) = \frac{p.q}{||p||||q||} = \frac{\sum_{i=1}^{n} p_i \times q_i}{\sqrt{\sum_{i=1}^{n}(p_i)^2} \times \sqrt{\sum_{i=1}^{n}(q_i)^2}} \qquad (6)$$

5.5. Clustering

A cluster-based classification scheme is proposed to speed up the classification process. Several recent advances have been made in Computer Vision by incorporating clustering algorithms for the canonicalization of large data sets: selecting exemplars, building unsupervised object recognizers, text on generation, and learning in low-level vision [26]. Classification can be performed in two steps: The training mode, where the feature vectors of learning samples are clustered first based on a certain criterion. Then the classification mode, where the distance between a test sample and every cluster is calculated, and the clusters that are nearest to the test sample are chosen as candidate clusters. Then the classes within those candidate clusters are selected as the candidates of the test sample [27]. Here, we have used two clustering methods- K-means and K-medoids for classification [20, 28].

5.6. K-Nearest Neighbor

The K-Nearest Neighbor Classifier is an efficient technique which is used when the classification problem has pattern classes that display a reasonably limited degree of variability. It considers each input pattern given to it and classifies it to a certain class by calculating the distance between the input pattern and the training patterns. It takes into account only k nearest prototypes to the input pattern during classification. The decision is generally based on the majority of class values of the k nearest neighbors. In the k-Nearest neighbor classification, we compute the distance between features of the test sample and the feature of every training sample [29].

5.7. Linear Classifier

Linear classifier is a statistical classifier which makes a classification decision based on the value of the linear combination of the features. A linear classifier is often used in situations where the speed of classification is an issue, since it is often the fastest classifier [25]. Linear classifiers often work very well when the number of dimensions in feature vector is large as represented by equation (7).

$$y = f(\vec{w}.\vec{x}) = f\left(\sum_j w_j x_j\right) \qquad (7)$$

where w_j is weight vector, learned from a set of labeled training samples. Xj is the feature vector of testing sample.

f is a simple function that maps the value to the respective classes based on a certain threshold.

5.8. Artificial Immune System

Artificial immune system (AIS) based classification approach is relatively new in the field of pattern recognition (PR). Artificial immune system (AIS) is inspired by mammalian immune system and has so far been applied in computer security, network intrusion detection, fraud detection, optimization, data analysis and clustering, machine learning, associative memories, fault and anomaly detection, control and scheduling, pattern recognition[30, 31]. The classification algorithm used here borrows major concepts from [30, 32] and some ideas from [33]. The algorithm has two stages: (i) training and (ii) classification. Training phase produces an immune memory (collection of antibodies or B-cells) that is used during classification stage [13].

5.9. Classifier Fusion

It is observed from the real life classification problems that usually the features are spread in many different ways and a common approach is to combine the selected features of different categories which are complementary into a single feature vector. However, when different types of features are combined into the same feature vector, some large-scaled features may dominate the distance, while the other features do not have the same impact on the classification [34]. Instead, separate classifiers can be used to classify based on each visual feature individually. The final classification can be obtained based on the combination of separate base classification results. Hence the non-homogenous properties of individual features do not necessarily affect the final classification directly. In this way, each feature has its own effect on the classification result. It has been found that a consensus decision of several classifiers can give better accuracy than any single classifier [35]. Thus, in the recent years combining classifiers has become a popular research area. The goal of combining classifiers is to form a consensus decision based on the opinions provided by different base classifiers. Combined classifiers have been applied to several classification tasks, for example to face recognition, handwritten character identification, and fingerprint verification [36], [37]. Here, we have used two feature extraction methods to extract the features. These features are classified using k-NN and linear classifiers. The outputs of these classifiers are fused to get the final decision [38].

6. Study of some Significant Factors

In this section, we describe the effect of changing certain parameters like feature extraction method, classifier, size of training data set *etc.*, on the performance of the recognition system. The different types of experiments carried out using different factors are as listed below:

- A combination of different feature extraction methods and classifiers.
- A single feature extraction method and different classifiers.
- A single classifier and different feature extraction methods.
- Variations in a single classifier. For example, changing the k value or distance metric in the k-NN Classifier.
- Variations in a single feature extraction method. For example, using different scales and orientations of the Curvelet Transform.
- Changing the size of the training data set.
- Combining the classifier decisions.

The effects of these factors have been studied to find out the better feature extraction method and classification technique among the different techniques that exist. We also use these results to find out feature extraction methods and classifiers which are less dependent on the training data and to find which methods give better recognition accuracy.

6.1. Recognition Accuracy of Different Feature Extraction Methods with Different Classifiers

To understand the effect of different features and classifier combinations we have conducted experiments using eight different features with nine classifiers. Such sixteen different combinations were used for experimentation and the results were tabulated as shown in the Table 1. It is clear from the results that the combination of zoning feature with artificial immune system gives a better result than the other combinations while the combination of image fusion technique with Manhattan distance metric gives least recognition rate. It also seen that on an average the directional features outperforms the other feature extraction methods with different classifier combinations proving that it is a good feature extraction method.

6.2. Recognition Accuracy of same Feature with Different Classifier

We have conducted experiments in order to find out which classifier gives efficient results with a particular feature extraction method. Here, we have tried using different feature extraction methods like directional features, Run length count and image fusion with different classifiers like K-NN,K- Means, K -Medoids, Linear classifier, Euclidean distance, Chebyshev, Manhattan and Cosine distance measures. First we experimented with the directional features by classifying with classifiers like K-NN, K- Means, K -Medoids, Linear classifier and Euclidean distance and the results are as shown in the Figure 4. It is seen from the results that the directional features with K-Medoids outperforms other combinations. Similarly run length count with K-NN and Image fusion with Euclidean distance perform better than other combinations.

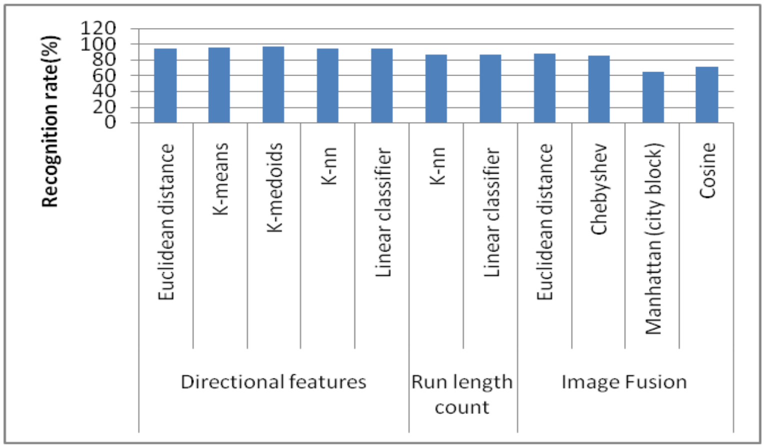


Figure 4. Recognition Accuracy of same Feature with Different Classifiers

Table 1. Recognition Accuracy of Different Feature Extraction Methods with Different Classifiers

Name of feature extraction method	**Size of feature vector**	**Classifier**	**Recognition Rate (%)**
Radon features	703	K-nn	90.4
Fanbeam features	360	K-nn	86
Directional features	72	Euclidean distance	94
		K-means	96
		K-medoids	97
		K-nn	94
		Linear classifier	94
Run length count	18	K-nn	87
		Linear classifier	86
Image Fusion	64	Euclidean distance	88
		Chebyshev	85.6
		Manhattan (city block)	64
		Cosine	71.2
Zoning	144	AIS	98.5
DFT	72	K-nn	89.33
Curvelet	348	K-nn	90.5

6.3. Recognition Accuracy of Different Features with same Classifier

Experiments were conducted to measure the performance of the classifiers with different feature extraction methods. Here, we have made an attempt to test the performance of K-nn classifier with Radon, Fanbeam, Directional, Run length count, DFT, Curvelet, Linear classifier with directional and run length features and Euclidean distance measure with image fusion and Directional features. The results are tabulated as shown in the Figure 5. It is clear from the results that K-NN, Linear classifier and Euclidean distance performs better with directional features then the other features.

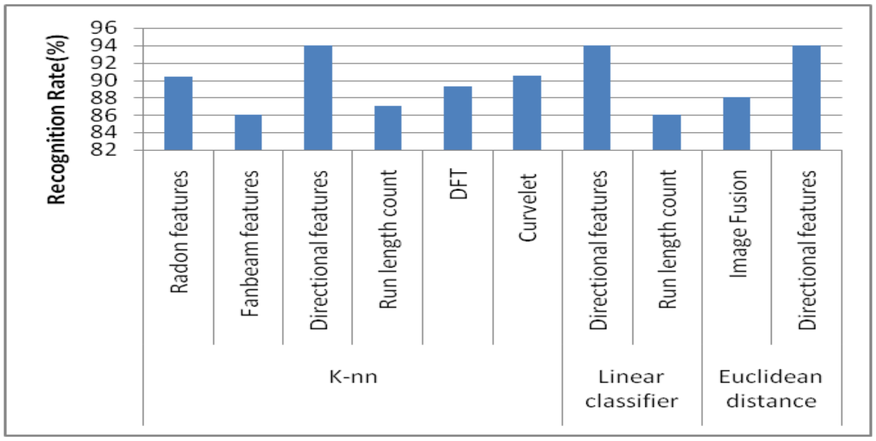


Figure 5. Recognition Accuracy of Different Features with same Classifier

6.4. Recognition Accuracy of Different Variations of the Feature Extraction Method with Different Options of the Classifier

A given feature may be used in different ways. We have tested the internal as well as external performance of each feature extraction method. The internal performance is considered by conducting the experiments on various possible variations for a given feature type. A feature extraction method may be used with various variations. It is essential to know the performance of several variations of a feature type. We have conducted experiments to predict the performance of different variations of each feature type to identify which feature vector among these variations is better. A single stage classification scheme has been used in all the experiments conducted here.

Similarly the internal performance of a classifier can be tested by conducting the experiments on various possible variations for a given classifier. A classifier may be used with the different variations, for example K-NN classifier where we can test the classifier by varying the value of K and by varying the distance measure used to find the nearest neighbor. We have conducted experiments to predict the performance of the different variations of K-NN classifier to identify which classification option from these variations is better. The experimental results using a single feature with different variations and single stage classifiers k-NN with different variations are given in Table 2.

The classification is done using Euclidean, Cosine, Cityblock and Correlation measures as the distance measures and nearest as the rule. Five different classifiers were used with the K values as 1,4,7,10,13. K value specifies the number of neighbors used for classification. From the results obtained, we can infer that it is not always necessary to use all the coefficients that are obtained by applying the Curvelet transform. Instead we can use the coefficients with larger values to form the feature vector. It can also be noted that the results have been independent of the change in the k value of the K-NN classifier. According to the Table II, for the testing samples when we use the information of scale 3 for achieving feature vector and Cityblock distance measure in classification, our recognition rate got better.

Table 2. Recognition Accuracy of Different Variations of the Feature Extraction Method with Different Options of the Classifier

Name of feature extraction method	Size of feature vector	classifier	Recognition Rate (%)
Curvelet with scale 1	171	K-nn with Euclidean distance	76.5
Curvelet with scale 2	426		75.55
Curvelet with scale 3	348		79.5
Curvelet with scale 4	322		83.5
Curvelet with scale 5	316		83
Curvelet with scale 1	171	K-nn with cosine distance	75.5
Curvelet with scale 2	426		75.5
Curvelet with scale 3	348		74
Curvelet with scale 4	322		77
Curvelet with scale 5	316		77
Curvelet with scale 1	171	K-nn with cityblock	81.5
Curvelet with scale 2	426		83
Curvelet with scale 3	348		90.5
Curvelet with scale 4	322		88.5
Curvelet with scale 5	316		89.5
Curvelet with scale 1	171	K-nn with correlation	76
Curvelet with scale 2	426		77
Curvelet with scale 3	348		74
Curvelet with scale 4	322		77
Curvelet with scale 5	316		76.5

6.5. Effect of Training Dataset Size on Recognition Accuracy

The training set size affects the recognition performance of a recognition system a lot. But a question arises, whether the discrimination ability of all the feature extraction methods is affected equally or is there little effect on the performance of some features, with small training sample set size, as compared to others? The performance of which feature type is least or mostly effected? To know the effects of training sample size on discrimination ability of various features and classifiers, we have conducted experiments by taking different sizes of training sample set [39].

For the experimentation, we have used two fold cross validation scheme for recognition result calculation. In fold-I 70% of the data is used for the training and in fold-II 80% of data is used for training. The recognition rates of first fold and second fold are averaged to obtain the average recognition rate. However better accuracy and time complexity is observed with 0% rejection rate when the classifier is trained with 80% of the data. Experimentation was conducted using directional chain code features and run length count. For classification, we have used linear classifier and K-nearest neighbor classifier. From the experimentation, we noted that the overall numeral recognition accuracy of the directional features is better than the run length count features. However the run length count feature uses only 18 features as compared to the 72 features of the directional features. For the directional features both the classifiers yielded same accuracy but the time complexity of the K-NN classifier was better than the linear classifier.

The analysis of effect of size of training data set on the recognition rate of directional and run length features using K-NN and Linear classifiers are given in Figure 6. From the figure, it is clear that the size of training data set affects the recognition rate of various features a lot but it is not uniform in case of all feature extraction methods. From the experimentation we noted that, the performance of run length features has a lot of effect whereas directional features have very small effect. That is, run length feature is highly dependent on the training data.

Similarly, the effect of training sample size on the recognition performance of k-NN and linear classifiers using two features, directional and run length has been analyzed in Figure 6. From the figure it is clear that the error rate of linear classifier with small sample size (70% training) is larger as compared to k-NN classifier.

The error rate with 80% training data size using run length and k-NN combination is 13% and has grown to 24% when the training data size is reduced to 70%. The difference in error rate is 11%. This difference in error rate is 10% for the run length and linear classifier combination. Similarly the difference in error rate for directional features with K-NN and linear classifier is 1.33% and 3.33% respectively. It means that the linear classifier is more dependent on training data as compared to K-NN.

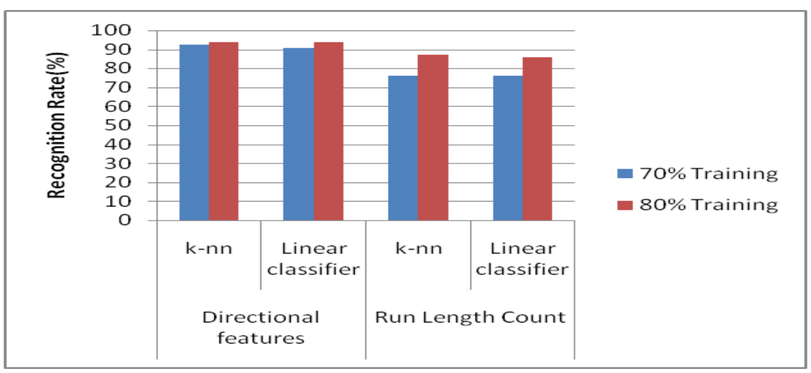


Figure 6. Effect of Training Data Set Size on Recognition Performance of Directional and Run Length Features with K-nn and Linear Classifiers

6.6. Effect of Fusing the Classifier Decision on Recognition Accuracy

The use of classifier combinations has been a subject of intensive research for the last ten years. Popular solutions on this field are bagging, boosting, and probability-based classifier combination. Voting is a simple combination strategy which is used when the probability distributions of the base classifiers are not available. In the voting-based classifier combinations, the majority of the base classifier outputs decide the final class of an unknown sample. Voting-based classifier combinations have been used in pattern recognition

To understand the effect of classifier fusion or combination we have conducted experiments by taking two feature extraction methods like directional chain code and run length code with k-NN classifier and linear classifier. Here, we have used k-NN and linear classifiers to classify based on Run Length Count and directional features individually and the recognition rates are tabulated as shown in table. It is seen from the experimentation that directional feature method misclassified the samples which are classified with run length feature method and vice versa. For example, we computed the accuracy of the individual numerals and observed that for the directional chain code feature, the lowest accuracy was obtained for the numeral 3. The common misclassification of numeral 3 was with numeral 7, which is very similar in shape. However the method was able to achieve good recognition rate for the other similar shaped numerals like 0 and 1, 6 and 9. Similarly for the RLC features, the lowest accuracy was obtained for the numeral 9. The common misclassification of numeral 9 was with numeral 2 and 7 even though they are not of similar shape. Then we applied classifier fusion method on same dataset where the final classification is obtained by combining the separate base classification results. It was found that the overall recognition accuracy improved when compared to any single classifier, which we had used. The classifier fusion method was able to take the advantage of both the methods. This is reflected in the results. The comparison of different classifiers with the classifier fusion method is shown in Figure 7.

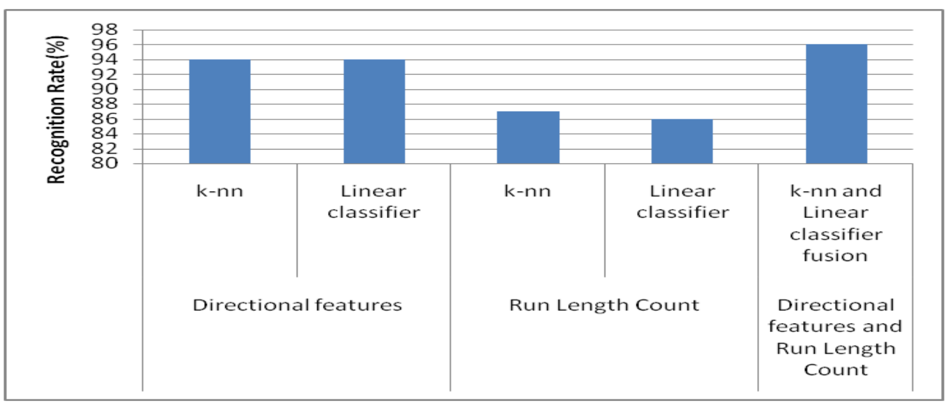


Figure 7. Effect of Fusing the Classifier Decisions on Recognition Accuracy

7. Comparative Study

The Table 3 shows the comparison results of existing methods with proposed method. From the comparative study it is seen that the proposed method gives better recognition accuracy than the other existing methods.

8. Discussions and Conclusion

To get an idea of the recognition results of different classifiers and feature extraction methods and to provide a new benchmark for future research a comparative study of handwritten Kannada numeral recognition with eight different type of features and nine different classifiers has been reported in this paper. Results of different classifiers with different feature extraction methods have been discussed here. It can be noted that the zoning features with Artificial Immune System outperforms the other combinations having a maximum recognition rate of 98.5%.

Among the various features considered here, on an average directional chain code based features gives a better result. We have also tried with different variations of the feature extraction method *i.e.*, curvelet and the different variations of the classifier i.e., k-NN by varying its K-value and the different distance measures. We have also tried with combining the classifiers and it is observed that from fusion there was an improvement in the recognition rate. Finally, we performed experiments to show the effect of training data set size. From the results it was observed that the 80% of the training data set size gave better results than the 70% of the training dataset size.

Table 3. Comparison of Proposed Method with the Existing Method

Authors	**No. of samples in data set**	**Feature extraction method**	**Classifier**	**Accuracy (%)**
B V Dhandra et al[40]	1512	Structural features	KNN classifer with 2nd order weighted Minkowski measure	96.12
Rajashekhara Aradhya S V et.al[41]	1000	Vertical projection distance with zoning	NN classifier	93
R Sanjeev Kunte et al[42]	2500	Wavelet	Neural classifier	92.3
G G Rajput et al[14]	1000	Image fusion	NN classifier	91.2
V N Manjunath Aradhya et al[43]	2000	Radon features	NN classifier	91.2
Dinesh Acharya U et. Al[44]	500	Structural features	K-means	90.5
Rajashekhara Aradhya S V et.al[45]	4000	Zoning with average angle from image centroid to each pixel in the zone	SVM	97.85
G G Rajput et al[46]	2500	Chain code an fourier descripter	SVM	97.34
Proposed Method	1000	Zoning	Artificial Immune System with K-nn (Hamming distance)	98.5

References

[1] L. M. Lorigo and V. Govindaraju, "Offline Arabic handwriting recognition: A Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 5, **(2006)**, pp. 712-724.

[2] N. Arica and F. T. Yarman-Vural, "An Overview of character recognition focused on off-line handwriting", IEEE Transactions on System, Man, and Cybernetics-Part C: Applications and Reviews, vol. 31, no. 2, **(2001)**, pp. 216-233.

[3] R. Plamondon and S. N. Srihari, "On-line and Off-line Handwritten Character Recognition: A Comprehensive Survey", IEEE. Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, **(2000)**, pp. 63-84.

[4] G. Nagy, "Chinese Character Recognition, a Twenty Five Years Retrospective", Proceedings of International Conference on Pattern Recognition, **(1988)**, pp. 109-114.

[5] U. Pal and B. B. Chaudhuri, "Indian Script Character Recognition: A Survey", Pattern Recognition, vol. 37, **(2004)**, pp. 1887-1899.

[6] M. Abdul Rahiman and M. S. Rajasree, "A Detailed Study and Analysis of OCR Research in South Indian Scripts", Proc. of International Conference on Advances in Recent Technologies in Communication and Computing, **(2009)**, pp. 31-38.

[7] G. G. Rajput, R. Horakeri and S. Chandrakant, "Printed and Handwritten Kannada Numeral Recognition Using Crack Codes and Fourier Descriptors Plate", IJCA Special Issue "Recent Trends in Image Processing and Pattern Recognition", RTIPPR, **(2010)**, pp. 53-58.

[8] T. Wakabayashi, U. Pal, F. Kimura and Y. Miyake, "F-ratio Based Weighted Feature Extraction for Similar Shape Character Recognition", Proceedings of the 10th International Conference on Document Analysis and Recognition, **(2009)**, pp. 456-460.

[9] G. G. Rajput and S. M. Mali, "Fourier descriptor based Isolated Marathi Handwritten numeral Recognition", International Journal of Computer Application, **(2010)**, pp. 5120/724-1017.

[10] S. V. Rajashekararadhya and P. Vanaja Ranjan, "Handwritten Numeral Recognition of Three Popular South Indian Scripts: A Novel Approach", Proceedings of the Second International Conference on Information Processing, **(2008)**, pp. 162-167.

[11] Ø. Due Trier , A. K. Jain and T. Taxt, "Feature extraction methods for character recognition-A survey", Pattern Recognition, vol. 29, no. 4, **(1996)**, pp. 641-662.

[12] A. V. Narasimha Murthy, "Kannada Lipiya Ugama Mattu Vikasa", Institute of Kuvempu Kannada Studies Publication University of Mysore, **(1975)**.

[13] H. R. Mamatha, K. Srikanta Murthy, K. S. Amrutha, P. Anusha and R. Azeemunisa, "Artificial Immune System based Recognition of Handwritten Kannada Numerals", Advanced Materials Research, ©(2012) Trans Tech Publications, Switzerland,doi:10.4028/www.scientific.net/AMR.433-440.900, vol. 433-440,**(2012)**, pp. 900-906.

[14] G. G. Rajaput and M. Hangarge, "Recognition of Isolated Handwritten Kannada Numerals Based on Image Fusion Method", PReMI07, LNCS.4815, **(2007)**, pp. 153-160.

[15] H. R. Mamatha, K. Srikanta Murthy, P. Vishwanath, T. S. Savitha, A. S. Sahana and S. Suma Shankari, "Evaluation of Similarity Measures for Recognition of Handwritten Kannada Numerals", CiiT International Journal of Digital Image Processing, ISSN 0974-9691 and Online: ISSN 0974-9586, DOI: DIP102011018, vol. 3,no. 16, **(2011)** October, pp. 1025-1029.

[16] M. Miciak, "Character recognition using Radon Transformation and Principal Component Analysis in postal applications", Proc. of International Multi conference on Computer Science and Information Technology, **(2008)** October 20-22, pp. 495-500.

[17] H. R. Mamatha, K. Srikanta Murthy, S. Sudan, V. G. Raj, S. S. Jois, "Fan Beam Projection Based Features to Recognize Handwritten Kannada Numerals", 2011 International Conference on Software and Computer Applications, IPCSIT © (2011) IACSIT Press, Singapore, vol. 9, **(2011)**.

[18] J. W. Beattie, "Tomographic Reconstruction from Fan Beam Geometry Using Radon's Integration Method", IEEE Transactions on Nuclear Science, vol. 22, no. 1, **(1975)** February, pp. 359-363.

[19] N. Sharma, U. Pal and F. Kimura, "Recognition of Handwritten Kannada Numerals", Proceedings of 9th International Conference on Information Technology (ICIT-2006), IEEE Computer Society Press, **(2006)**, pp. 133-136.

[20] H. R. Murthy Mamatha, K. Srikanta Veeksha, A. V. Vokuda and M. P. S. Lakshmi, "Recognition of Handwritten Kannada Numerals Using Directional Features and K-Means", IEEE International Conference on Computational Intelligence and Communication Networks (CICN), **(2011)**, pp. 644-647.

[21] S. Karthik, H. R. Mamatha and K. Srikanta Murthy, "A DFT Based Algorithmic Approach To Recognize The Handwritten Kannada Numerals", CSI sponsored National Conference on Emerging Trends in Information and Communication technologies, Hyderabad, **(2012)** February 3-4.

[22] S. Karthik, H. R. Mamatha and K. Srikanta Murthy, “Kannada Characters Recognition-A Novel Approach Using Image Zoning and Run Length Count”, CIIT International journal of digital image processing, vol. 3, **(2011)**, pp. 1059-1062.
[23] F. Mohamad Kazemi, J. Izadian, R. Moravejian, E. Mohamad Kazemi, “Numeral Recognition Using Curvelet Transform”, ACS International Conference on Computer Systems and Applications, **(2008)**, pp. 606-612.
[24] H. R. Mamatha, S. Sucharitha and K. Srikanta Murthy, “Handwritten Kannada Numeral Recognition based on the Curvelets and Standard Deviation”, IEEE sponsored 4th International Conference on Electronics Computer Technology, **(2012)** April 6-8, pp. 185-189.
[25] The Wikipedia website.[Online].Available: http://www.wikipedia.org/.
[26] A. Fitzgibbon and A. Zisserman, “On Affine Invariant Clustering and Automatic Cast Listing in Movies”, Proceedings of 7th European Conference on Computer Vision, ECCV, vol. 3, **(2002)**, pp. 304-320.
[27] M. Nadler and E. P. Smith, “Pattern Recognition Engineering”, John Wiley & Sons, New York, **(1993)**.
[28] H. R. Murthy Mamatha, K. Srikanta Veeksha, A. V. Vokuda, P. S. Lakshmi, “Recognition of Hand written Kannada numerals using K-Medoids”, Advanced Materials Research, © (2012) Trans Tech Publications, Switzerland, vol. 433-440, **(2012)**, pp. 5354-5358.
[29] S. Theodoridis and K. Koutroumbas, “Pattern Recognition”, Academic Press, New York, **(1999)**.
[30] J. Timmis, “Artificial Immune Systems: a novel data analysis techniques inspired by the immune network theory”, PhD Thesis, University of Wales, Aberystwyth, **(2001)**.
[31] D. Dasgupta, “An overview of artificial immune systems and their applications”, Artificial Immune Systems and their Applications, Springer, **(1998)**, pp. 3-21.
[32] A. B. Watkins, “AIRS: A Resource Limited Artificial Immune Classifier”, Master’s dissertation, Dept. of Computer Science, Mississippi State University, **(2001)**.
[33] J. H. Cater, “The Immune System as A Model for Pattern Recognition and Classification”, Journal of the American Medical Informatics Association, vol. 7, no. 1, **(2000)**, pp. 28-41.
[34] R. O. Duda, P. E. Hart and D. G. Stork, “Pattern Classification”, 2nd ed., John Wiley & Sons, New York, **(2001)**.
[35] J. Kittler, M. Hatef, R. P. W. Duin and J. Matas, “On Combining Classifiers”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 20, no. 3, **(1998)**, pp. 226-239.
[36] L. Xu, A. Krzyzak and C. Y. Suen, “Methods for Combining Multiple Classifiers and their Applications to Handwriting Recognition”, IEEE Trans. on Systems, Man and Cybernetics, vol. 22, no. 3, **(1992)**, pp. 418-435.
[37] A. K. Jain, S. Prabhakar and S. Chen, “Combining Multiple Matchers for a High Security Fingerprint Verification System”, Pattern Recognition Letters, vol. 20, no. 11-13, **(1999)**, pp. 1371-1379.
[38] H. R. Mamatha, S. Karthik and K. Srikanta Murthy, “Classifier Fusion Method to Recognize Handwritten Kannada Numerals”, IEEE sponsored 4th International Conference on Electronics Computer Technology, **(2012**) April 6-8, pp. 640-644.
[39] H. R. Mamatha, S. Karthik and K. Srikanta Murthy, “Feature Based Recognition of Handwritten Kannada Numerals-A Comparative Study”, IEEE sponsored International Conference on Computing, Communications and Applications, **(2012)** February 22-24, pp. 1-6.
[40] B. V. Benne Dhandra, R. G. Hangarge, “Handwritten Kannada Numeral Recognition Based on Structural Features”, Conference on Computational Intelligence and Multimedia Applications, **(2007)**.
[41] S. V. Rajashekararadhya and P. Vanaja Ranja, “Neural Network Based Handwritten Numeral Recognition of Kannada and Telugu scripts”, TENCON, **(2008)**.
[42] R. Sanjeev Kunte and R. D. Sudhakar Samuel, “Script Independent Handwritten Numeral Recognition”, VIE -2006, **(2006)**, pp. 94-98.
[43] V. N. Manjunath Aradhya, G. Hemanth Kumar and S. Noushath, “Robust Unconstrained Handwritten Digit Recognition Using Radon Transform”, Proc. of IEEE ICSCN 2007, **(2007)**, pp. 626-629.
[44] U. Dinesh Acharya, N. V. Subba Reddy and Krishnamurthy, “Isolated handwritten Kannada numeral recognition using structural feature and K-means cluster”, IISN, **(2007)**, pp. 125-129.
[45] S. V. Rajashekararadhya and P. Vanaja Ranjan, “Support Vector Machine based Handwritten Numeral Recognition of Kannada Script”, 2009 IEEE International Advance Computing Conference, **(2009)** March 6-7.
[46] G. G. Rajput and R. Horakeri, “Shape descriptors based handwritten character recognition engine with application to Kannada characters”, 2nd International Conference on Computer and Communication Technology (ICCCT), **(2011)**.

Authors

Mamatha H. R. received her B E degree in Computer Science and Engineering from the Kuvempu University in 1998.and the M.Tech degree in Computer Networks and Engineering from the Visvesvaraya Technological University in 2006.Since 2008 she has been a Ph.D. student at the Visvesvaraya Technological University. She has 14 years of teaching experience. Currently she is working as an Assistant Professor in the Department of Information Science and Engineering, P E S Institute of Technology. Her current research interests include pattern recognition and image processing. She has published 14 papers in various Journals and Conferences of International repute. She is a life member of Indian Society for Technical Education. She has mentored students for various competitions at international level.

Srikanta Murthy K. received B.E (Electrical and Electronics Engineering). degree in 1986, M.Tech (Power Systems) in 1996 from National Institute of Engineering, University of Mysore and Ph.D. degree in Computer Science from University of Mysore in 2006. He joined the faculty of Electrical Engineering in NIE in 1987 and worked for K V G College of Engineering at various positions from 1991-2004. Presently he is working as Professor and Head, Department of CS&E, P E S School of Engineering, Bangalore, India. He has served as Chairman Board of Examiners in computer science in Mangalore University and also the member of local inspection committee, Mangalore University and Visvesvaraya Technological University. Currently he is guiding 5 candidates towards the doctoral degree and also guided many projects at PG level. His current research interests are computer vision, image processing and pattern recognition. He has published 60+ papers in various Journals and Conferences of International repute. He is a life member of Indian Society for Technical Education, Indian Society of Remote Sensing, Computer Society of India, Society of Statistics and Computer Applications and associate member of Institute of Engineers.