# Handwritten Digit Recognition Using K-Nearest Neighbour Classifier

U Ravi Babu
Research Scholar,
Aacharya Nagarjuna University,
Assoc. Professor – GIET,
Rajahmundry, A.P, India.
uppu.ravibabu@gmail.com

Dr. Y Venkateswarlu
Professor & Head,
Department of CSE,
GIET Engg College,
Rajahmundry, A.P, India.
yalla_venkat@yahoo.com

Aneel Kumar Chintha
M.Tech (CSE) Student,
GIET, Rajahmundry,
A.P, India.
aneelkumar.chintha@gmail.com

***Abstract*— This paper presents a new approach to off-line handwritten digit recognition based on structural features which is not required thinning operation and size normalization technique. In this paper uses four different types of structural features namely, number of holes, water reservoirs in four directions, maximum profile distances in four directions, and fill-hole density for the recognition of digits. The digit recognition system mainly depends on which kinds of features are used. The main objective of this paper is to provide efficient and reliable techniques for recognition of handwritten digits. A Euclidean minimum distance criterion is used to find minimum distances and k-nearest neighbor classifier is used to classify the digits. A MNIST database is used for both training and testing the system. 5000 images are used to test the proposed method a total 5000 numeral images are tested and got 96.94% recognition rate.**

***Keywords-Structural features, fill hole density, digit recognition, profile distance, handwritten digits***

## I. INTRODUCTION

Offline handwritten character recognition is one of the practically important issues in pattern recognition applications. The applications of digit recognition includes in postal mail sorting, bank check processing, form data entry, etc. In above application digit recognition is important to the overall system in terms of performance (accuracy and speed). In recent years, different algorithms and classification approaches have been presented to solve this problem[1]. Digit recognition is an important component of handwritten character recognition system due to it's vide application. From more than three decades can achieve high classification high recognition rates in the area of recognition of handwritten numerals [2].

The feature extraction and the classification technique plays an important role in offline character recognition system performance. Various feature extraction approaches have been proposed for character recognition system [14]. The problems faced in handwritten numeral recognition has been studied while using the techniques like Dynamic programming, HMM, neural network, Knowledge system and combinations of above techniques [15]. Wide-ranging work has been carried out for digit recognition in so many languages like English, Chinese, Japanese, and Arabic. In Indian mainly worked in Devanagari, Tamil, Telugu and Bengali numeral recognition [16, 17].

Recognition of handwritten English Numerals is complicated task due to its unconstrained shapes, writing style variation from person to person and different kinds of noise presence that break the continuity in the numerical character and change in their topology. The feature extraction plays major role in numerical recognition system. Numbers of feature extraction methods are stated in the literature, like template matching, projection histogram, zoning, and various moment techniques [18] to enable for specific applications. Some methods include fuzzy features [7, 8], invariant moments features [8], template and deformable Templates [11, 12], structural and statistical features [9, 8] extraction. In B.V.Dhandra et.al proposed method uses 13 structural features are extracted and recognition rate is calculated only on 1512 test images tested and achieved 96.12%.

In this paper proposed a new technique for feature extraction based on the maximum profile distance [4] The feature set includes number of holes in an image, Water Reservoir principle based features [10], Maximum profile distances and filling hole density. Totally ten features are extracted: number of connected components is the first feature, four water reservoir features, four maximum profile density and one fill hole density feature. For calculating accuracy of the proposed method 50000 images are used for training set and 5000 images are used for test set

The paper is organized as follows; Section 2 contains Database and the preprocessing. Feature extraction method is described in Section 3. The proposed algorithm is presented in Section 4. The in Classifications method is in Section 5. The experimental details and results obtained are presented in Section 6. Section 7 contains the conclusion part.

## II. DATABASE AND PREPROCESSING

### A. Database

The proposed method uses MNIST (Modified NIST) [6] database which includes a training set of 60,000 images and a test set of 5000 images. The training and test sets are subset of NIST digit base. The MNIST digit database contained fixed size images and digit image (foreground

978-1-4799-2877-4/14 $31.00 © 2014 IEEE
DOI 10.1109/WCCCT.2014.7

pixels) is center alignment with respect to the background pixels. The MNIST digit database is good database for applying learning techniques and patterns recognition methods because of this database need less time for noise removal in preprocessing. Originally, The MNIST database was constructed from NIST's Special Databases (SD) 3 and 7 which contain grayscale images of handwritten digits. The SD-3 was collected from employees of Census Bureau and SD-7 images were collected local high-school students. The MNIST training set is composed of 30000 images from SD-3 and 30000 images from SD-7. Totally, 60000 images are taken from 750 writers. However, the two databases, were written by totally different sources of writers and show different styles. The sample binary images were normalized into 20×20 gray-scale images with aspect ratio preserved and the normalized image is located in a 28 × 28 plane. The normalized image data are available at the homepage of LeCun [6]. Some images are shown in Fig. 1.

In this paper uses a set of 25000 images from SD-3 and 25000 images from SD-7 for training set and a set of 2500 images from SD-3, 2500 images from SD-7 for test set. The training set and test set were disjoint sets. The figure 1 shows the samples of the numerals of MNIST database.

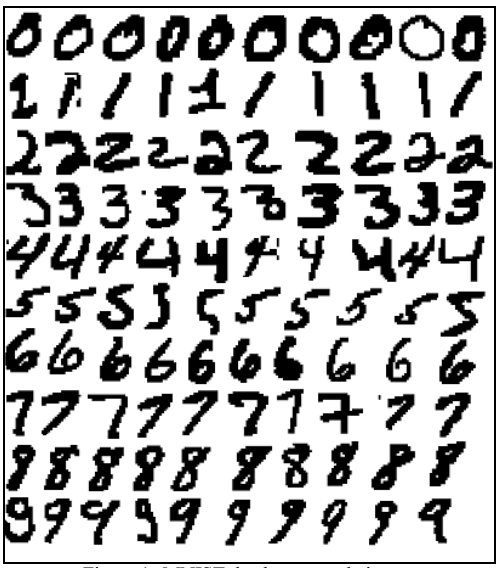

Figure 1. MNIST database sample images.

*B. Preprocessing the Binary image*

The recognition of handwritten numerals can achieve high performance based on pre processing stage also. In MNIST handwritten digit database images are in gray scale images, convert images in database into binary images. Converting the gray level images into binary based on the threshold value. After convert the images into binary, these images may have surplus elements one's (black) at undesirable places in the background image is called noise. It is necessary to remove noise from the image. To remove these unwanted one's from the background, need an algorithm. In this algorithm 3X3 template is used. It is assumed that the pixel (p, q) as center pixel and neighbors of the point (p, q) are (p- 1, q), (p-1, q+1), (p, q+1), (p+1, q+1), (p+1, q), (p+1,q-1), (p, q-1), and (p-1, q-1), as is shown in Figure 2.

| | | |
|---|---|---|
| **p-1, q-1** | **p-1, q** | **p-1, q+1** |
| **p, q-1** | **p, q** | **p, q+1** |
| **p+1, q-1** | **p+1,q** | **p+1, q+1** |

Figure 2. 8-neighbors of a **3x3** window

Noise removal in an image depends on the type of noise in that image. To remove the surplus elements in MNIST database, the patterns which are shown in figures 3, 4, 5, and 6 are well suited. Scan the image from first row fist column consider the 3 rows, 3 columns, and compare the those values with templates which are defined in figures 3,4,5 and 6. If pixel (p,q) is consider as a center pixel and that value is one and eight neighbors are zero (shown in figure 3) in the background then pixel (p,q) value is changed to zero, that concept is consider as **Isolated Pixel Removal**. If pixel (p,q) is consider as a center pixel and that value is one and any one of the neighbor(shown in figure 4) is also one then convert the center pixel and neighboring pixel which contains the value one to zero, that concept is consider as **two pixel width noise removal**. If pixel (p,q) is consider as a center pixel and that value is one and any two of the neighbors (shown in figure 5) are also one then convert the center pixel and neighboring pixels which contains the value one to zero, that concept is consider as **three pixel width noise removal**. If pixel (p,q) is consider as a center pixel and that value is one and any three of the neighbors (shown in figure 6) are also one then convert the center pixel and neighboring pixels which contains the value one to zero, that concept is consider as **four pixel width noise removal.** The results are shown in Figure 7. Results of the preprocessing step are shown in figures 7 and 8.

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

Figure 3. Isolate pixel pattern.

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 1 | 0 |
| 0 | 0 | 0 |

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 1 | 0 |

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 1 |

Figure 4. Two pixel patterns for noise removal

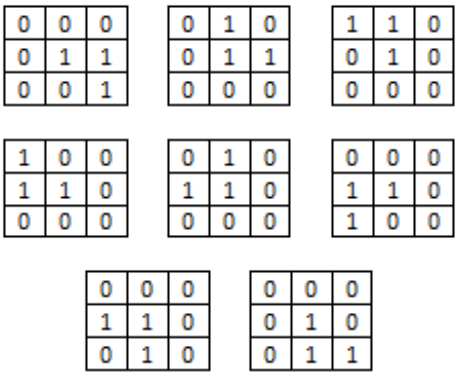


Figure 5. Three pixel patterns for noise removal

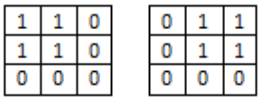


Figure 6. Four pixel patterns for noise removal

1) Results of preprocessing:

Figure 7. Before preprocessing numerical digits

Figure 8. After preprocessing numerical digits

## III. FEATURE EXTRACTION METHOD

Feature extraction is an important phase of any recognition system and in particular numerical recognition. In this paper, structural features are used for the recognition of numerals. The number of loops in the image (1 feature), water Reservoir principle based features (4 features), maximum profile distances (4 features) and fill hole density feature (1 feature) are used for the numeral recognition. Totally 10 features are extracted from each image.

### *A. Number of loops in a image*

The number of digit image loops is a structural feature. So many techniques are available to find the number of loops in an image. The connected component labeling algorithm [13] is used to find the number of loops. The number of background components (white components) minus one is the number of loops. For example, digits 0, 6, 9 have one loop because it has two background components and digit 8 has two loops. The large background component surrounding the digit (always present) and the small component enclosed within the loop.

### *B. Water reservoir features*

The water reservoir principle is as follows. If water is dispensed from a side of a component, the cavity regions of the component where water will be stored are considered as reservoirs [3]. The opening regions of the components where water will be stored are considered as reservoir. Handwritten digits generate reservoirs which are used for classification. Now, It will discuss the water reservoir extraction scheme of Handwritten Digits.

***Top reservoir:*** By top reservoirs of a digit, it means the reservoirs obtained when water is poured from top of the digit. The water reservoir area is appeared when digit image is unconnected. The figure 9 shows Top reservoir area of the Digit4. If digits are unconnected then the cavity regions are generated. Figure 10 illustrates the unconnected digits of 0, 6, 8 and 9


Figure 9. Top reservoir area of the digit 4


Figure 10. Unconnected digit that causes generate the cavity regions

***Bottom reservoir:*** By bottom reservoirs of a digit, it means the reservoirs obtained when water is poured from bottom of the digit. A bottom reservoir of a digit is visualized as a top reservoir when water will be poured from top after rotating the digit image by 180 degrees. Generally bottom reservoir is not obtained for any image except if the image is slant towards the bottom. The figure 11 shows that illustrations

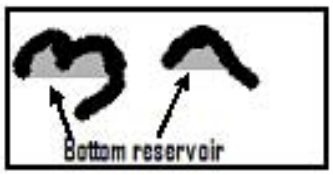


Figure 11. Bottom reservoirs of the digits 3 and 7 when they slant to bottom

***Left (right) reservoir*:** If water is poured from the left (right) side of a component, the cavity regions of the digit where water will be stored are considered as left (right) reservoirs. The figure 12 and 13 illustrates the left and right reservoirs.


Figure 12. Left reservoirs of the digits 3 and 5


Figure 13. Right reservoirs of the digits 6 and 4

The ratios of the Water Reservoir of the pixels with the total area are computed in four directions and they are stored as the feature vector. From the figure 9 - 13 it can be observed that digits 0 and 1 does not obtain the water reservoir area in any side, but if the digit 0 is unconnected then Top reservoir may be obtained that shows in Fig 10. Generally, the digit 2 generates left and right reservoir area. If digit image like shown in figure 14a then bottom reservoir area is also obtained for digit 2. Water reservoir area is shaded in figure 14a. For digit 3 left and right profiles are obtained if there is any slant towards the top (bottom), top (bottom) water reservoir area is obtained that shows in figure 14b. In digit 4 obtain top reservoir area. If there is any slant in digit then right reservoir is also occurred. If digit 4 is closed component then reservoir may or may not occur that shows in below figure 15a. In digit 5, left and right reservoir areas are occurred. If there is any slant in the image then reservoir area may be changed, but in opposite sides. If digit is unconnected, only one side only reservoir area is occurred that illustration shown in figure 15b. For the digit 6, right side reservoir area is obtained that shows in figure 13. If digit 6 unconnected, top reservoir area is also obtained, that illustrations show in figure 15c. For digit 7, left side reservoir area is obtained. If there is slant bottom reservoir is obtained that shows in figure 11. For digit 8 left and right sides reservoir area is occurred. If digit8 image is unconnected then top reservoir area is also occurred that shows in figure 10. For the digit 9, left side and sometimes bottom reservoir area is occurred. If digit 9 is unconnected then top and either left or bottom area is occurred that shows in figure 10.

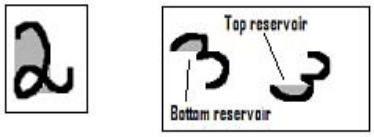


Fig 14a Fig 14b

Figure 14. Reservoir areas of 2 and 3 digit images


Fig. 15a Fif. 15b


Fig. 15c

Figure 15. Reservoir areas of 4, 5 and 6 digit images

*C. Maximum profile distance*

After fitting the bounding box on each numeral image, image profiles are computed in four directions. While computing the profile, This method considered only 40% of the middle area in four directions of the bounding box. Thus the maximum profile is obtained in four directions, the profile feature computations are illustrated in figure 16

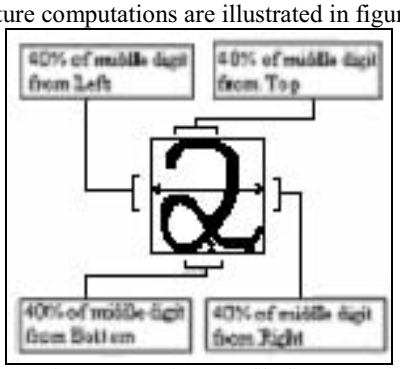


Figure 16. Maximum profile distances

*D. Fill Hole density*

A hole is a set of image pixels (generally foreground pixels) touches the border in a way that divides the background into two regions. To avoid such that needs to fill specific area to connectivity of background pixels. The proposed method uses 4-connected neighborhood connectivity rule for connectivity of background pixels. The 4-connectivity pixels can be defined as the pixels are neighbors that touches center pixel and these pixels are connected horizontally and vertically to the center pixel in 3×3 sub window of an image. The 3×3 window consists a set of 0's and 1's. In 3×3 window the center pixel position is 2×2. The looping area of the numeral is filled with ON pixels [13], further the ratio of fill-hole density with total area is estimated and taken as a feature.

## IV. ALGORITHM

The overall classification design of the MNIST digit database is shown in following algorithm.

**Algorithm: Classification of Digits**

**Input**: Isolated Numeral images from MNIST Database

**Output**: Recognition of the Numerals

**Method:** Structural features and K-*nn* classifier.

Step1: Convert the gray level image into Binary image

Step2: Preprocessing the Binary Image

Step3: Perform region labeling for each input image

Step4: Find the no of loops in image based region labeling and it is treated as a feature vector

Step5: Find Water Reservoir principle based features in four directions

Step6: Fit a minimum rectangle bounding box and crop the image.

Step7: Compute the maximum profile distances from all sides of bounding box.

Step8: Repeat the steps from 1 to 7 for all images in the Sample and Test Database.

Step9: Estimate the minimum distance between feature vector and vector stored in the library by using Euclidian distances.

Step10: Classify the input images into appropriate class label using minimum distance K-nearest neighbor classifier.

**End**

## V. RESULTS

The above mentioned algorithm uses 50000 images for training and 5000 images testing. The feature vector of 30 training images are listed out in table1. In table Left, Right, Top and Bottom is represented by L, R, T and B respectively.

TABLE I. SAMPLE FEATURE VECTOR OF MNIST DIGIT DATABASE

| Sno | Digit | No of Holes | Water Reservoir count | | | | Maximum profile desity | | | | FILL DESITY |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | R | T | B | L | R | T | B | |
| 1 | 5 | 0 | 50 | 33 | 0 | 0 | 10 | 10 | 1 | 16 | 0 |
| 2 | 0 | 1 | 0 | 0 | 0 | 0 | 8 | 8 | 7 | 6 | 72 |
| 3 | 4 | 0 | 0 | 0 | 102 | 5 | 14 | 3 | 10 | 8 | 0 |
| 4 | 1 | 0 | 0 | 0 | 0 | 0 | 9 | 8 | 12 | 13 | 0 |
| 5 | 9 | 1 | 0 | 20 | 0 | 0 | 8 | 4 | 4 | 10 | 10 |
| 6 | 2 | 1 | 28 | 0 | 0 | 6 | 12 | 10 | 12 | 4 | 3 |
| 7 | 1 | 0 | 86 | 0 | 0 | 0 | 2 | 3 | 13 | 11 | 0 |
| 8 | 3 | 0 | 59 | 1 | 0 | 0 | 12 | 7 | 14 | 11 | 0 |
| 9 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 1 | 4 | 0 |
| 10 | 4 | 0 | 6 | 3 | 38 | 125 | 15 | 9 | 10 | 10 | 0 |
| 11 | 3 | 0 | 167 | 0 | 0 | 0 | 9 | 4 | 16 | 2 | 0 |
| 12 | 5 | 0 | 0 | 4 | 0 | 0 | 12 | 12 | 12 | 11 | 0 |
| 13 | 3 | 0 | 92 | 3 | 0 | 0 | 15 | 5 | 9 | 18 | 0 |
| 14 | 6 | 1 | 0 | 27 | 0 | 0 | 8 | 9 | 10 | 5 | 8 |
| 15 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 16 | 7 | 0 | 51 | 0 | 0 | 0 | 9 | 10 | 2 | 14 | 0 |
| 17 | 2 | 2 | 209 | 0 | 150 | 0 | 11 | 10 | 10 | 6 | 8 |
| 18 | 8 | 2 | 2 | 0 | 0 | 0 | 8 | 9 | 12 | 14 | 4 |
| 19 | 6 | 0 | 0 | 40 | 0 | 0 | 6 | 7 | 6 | 1 | 34 |
| 20 | 9 | 1 | 8 | 0 | 0 | 0 | 6 | 9 | 16 | 14 | 6 |
| 21 | 4 | 0 | 0 | 0 | 64 | 11 | 14 | 2 | 14 | 12 | 0 |
| 22 | 0 | 1 | 0 | 0 | 0 | 0 | 11 | 6 | 8 | 4 | 69 |
| 23 | 9 | 1 | 7 | 0 | 0 | 0 | 5 | 8 | 4 | 10 | 17 |
| 24 | 1 | 0 | 0 | 0 | 0 | 0 | 9 | 10 | 12 | 14 | 0 |
| 25 | 1 | 0 | 44 | 11 | 0 | 0 | 9 | 4 | 15 | 1 | 0 |
| 26 | 2 | 0 | 47 | 46 | 0 | 0 | 12 | 12 | 14 | 2 | 0 |
| 27 | 4 | 0 | 0 | 4 | 31 | 0 | 7 | 4 | 8 | 10 | 0 |
| 28 | 3 | 0 | 69 | 223 | 0 | 0 | 12 | 4 | 2 | 5 | 0 |
| 29 | 2 | 0 | 56 | 26 | 0 | 0 | 13 | 7 | 3 | 17 | 0 |
| 30 | 7 | 0 | 23 | 0 | 0 | 0 | 7 | 10 | 2 | 9 | 2 |

## VI. CLASSIFICATION

The proposed method uses ***k*-nearest neighbor (k-nn) classification** algorithm for classifying the MNIST digit images in test database using the feature vector of training database. The ***k*-nearest neighbor algorithm** (*k*-NN) is classification technique to classify the objects base on training features space. The functionality of *k*-NN algorithm is to define the computations until classification is done irrespective of the learning techniques. Generally k-NN has two learning techniques. They are instance-based and lazy learning techniques. *K-nearest* neighbor algorithm is simplest classification technique because of computations are simple. The classification of objects based on votes of its neighbors which represented by by k. In K-nn object is classified to a particular class which has majority of votes.

In the k-Nearest neighbor classification, In K-nn compute the distance between feature values of the test sample and the feature vector values of every training image and The class of majority among the k-nearest training samples is based on the Euclidian distance measures. The training vector is a multidimensional array. Each row in an array contains feature values and corresponding class label of the training images where as test vector contains only feature values. In classification process, for each row in test vector assign the class label based on the Euclidian distance measures and number of neighbors (k) considered. The k value is defined by the user.

The algorithm is executed with the value of k is 1, 3, and 5. The graphical representation of the accuracy of classification in using various k values are shown in figure 17 and the overall classification results are listed out in table 2. From table 2 and figure 17 it is clearly evident that the optimal value of k is 1 for classification of MNIST numerical digits by using k-nearest neighbor classification technique. The recognition rate of the individual digits in test samples by using *k*-nearest neighbor classification algorithm (with k value 1) are listed in table 3 and from that table the overall recognition rate of the test database is 96.94%.

TABLE II. ACCURACY RATE USING DIFFERENT VALUES OF K WITH KNN CLASSIFIER

| NN classifiers with different K values | Number of test samples | Accuracy in percentage |
|---|---|---|
| K=1 | 5000 | 96.94 |
| K=3 | 5000 | 91.63 |
| K=5 | 5000 | 85.56 |

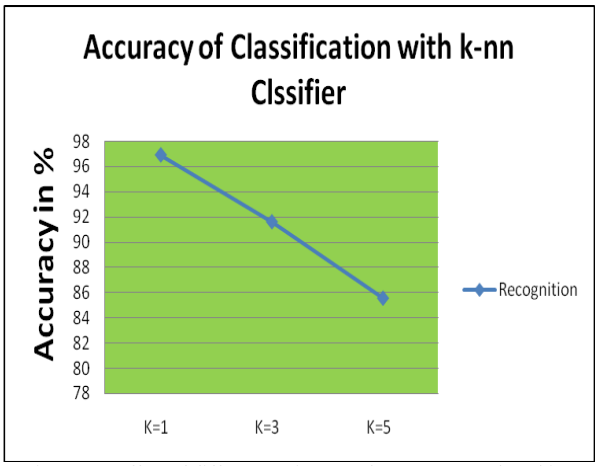


Figure 17. Effect of different testing samples on accuracy by taking different values of k

TABLE III. PERCENTAGE OF RECOGNITION FOR 5000 TEST IMAGES

| Digit | Test Images | Correctly not Classified | Correctly Classified | % Accuracy |
|---|---|---|---|---|
| 0 | 425 | 15 | 410 | 96.47 |
| 1 | 635 | 13 | 622 | 97.95 |
| 2 | 585 | 16 | 569 | 97.26 |
| 3 | 532 | 13 | 519 | 97.56 |
| 4 | 550 | 15 | 535 | 97.27 |
| 5 | 434 | 11 | 423 | 97.47 |
| 6 | 435 | 17 | 418 | 96.09 |
| 7 | 491 | 21 | 470 | 95.72 |
| 8 | 443 | 19 | 424 | 95.71 |
| 9 | 470 | 13 | 457 | 97.23 |
| **Total** | **5000** | **153** | 4847 | 96.94 |

## VII. CONCLUSIONS

In this paper used ten (10) structural features for recognition of handwritten numerals. In any recognition process, the important problem is to address the feature extraction and correct classification approaches. The proposed algorithm tries to address both the factors and well in terms of accuracy and time complexity. The Overall accuracy of 96.94% is achieved in the recognition process. The novelty of this method is that, it is thinning free, free from size normalization, accurate, and independent of digit size and writer style/ink independent, fast and accurate. This work is carried out as an initial attempt, and the aim of the paper is to facilitate for robust Telugu OCR. It is our future Endeavour to modify this algorithm and design a still robust handwritten Telugu OCR for high recognition rate and also recognition of offline handwritten digit recognition with less number of features and without using any standard classification algorithm.

## REFERENCES


[1] A hybrid method for unconstrained handwritten numeral recognition by combining structural and neural ``gas'' classifiers.

[2] S. Mori, C.Y. Suen, and K. Yamamoto. "Historical Review of OCR Research and Development," *Proceedings of the IEEE,* vol. *80,* no. 7, pp. 1029-1057, 1992.

[3] U Pal and P.P.Roy, "Multi-oriented and curved text lines extraction from Indian documents", IEEE Trans on system, Man and Cybernetics-Part B, vol.34, pp.1667-1684, 2004.

[4] B.V.Dhandra, R.G.Benne, Mallikarjun Hangarge, " Handwritten Kannada Numeral Recognition Based on Structural Features" Int. conference on Computational Intelligence and multimedia Applications 2007.

[5] Y. LeCun, et al., Comparison of learning algorithms for handwritten digit recognition, in: F. Fogelman-Souli e, P. Gallinari (Eds.), Proceedings of the International Conference on Artificial Neural Networks, Nanterre, France, 1995, pp. 53–60.

[6] Yann LeCun,"THE MNIST database of handwritten digits" Courant Institute, NYU Corinna Cortes, Google Labs, NewYork .http://www.research.att.com/yann/exdb/mnist/index.html

[7] Shamic Surel, P.K.Das, "Recognition of an Indian Scripts Using Multilayer Perceptions and fuzzy Features" Proc. Of 6th Int. Conf. on Document Analysis and Recognition (ICDAR), Seattle, pp 1220-1224, 2001.

[8] P.Nagabhushan, S.A.Angadi, B.S.Anami, "A fuzzy statistical approach of Kannada Vowel Recognition based on Invariant Moments", Proc. Of NCDAR-2003, Mandy, Karnataka, India, pp275-285, 2003.

[9] L.Heutte, T.Paquest, J.V.Moreau, Y.Lecourtier, C.Oliver, "A structural/ statistical feature based vector for handwritten character recognition", Pattern Recognition, p.629-641, 1998.

[10] U Pal and P.P.Roy, "Multi-oriented and curved text lines extraction from Indian documents", IEEE Trans on system, Man and Cybernetics-Part B, vol.34, pp.1667-1684, 2s004.

[11] J.D. Tubes, A note on binary template matching. Pattern Recognition, 22(4):359-365, 1989.

[12] Anil K.Jain, Douglass Zonker, "Representation and Recognition of handwritten Digits using Deformable Templates", IEEE, Pattern analysis and machine intelligence, vol.19, no-12, 1997.

[13] R.C.Gonzal, R.E.Woods, "Digital Image Processing", Pearson Education, 2002

[14] O.D. Trier, A.K. Jain, T. Taxt, Feature extraction methods for character recognition—a survey, Pattern Recognition 29 (4) (1996) 641–662.

[15] A.L.Koerich, R. Sabourin, C.Y.Suen, "Large off-line Handwritten Recognition: A survey", Pattern Analysis Application 6, 97-121, 2003.

[16] A.F.R. Rahman, R.Rahman, M.C.Fairhurst, "Recognition of handwritten Bengali Characters: A Novel Multistage Approach", Pattern Recognition, 35,997-1006, 2002.

[17] R. Chandrashekaran, M.Chandrasekaran, Gift Siromaney, "Computer Recognition of Tamil, Malayalam and Devanagari characters", Journal of IETE, Vol.30, No.6, 1984.

[18] Oivind Trier, Anil Jain, Torfiinn Taxt, "A feature extraction method for character recognition-A survey "", pattern Recg, vol 29, No 4, pp-641-662, 1996