Printed and Handwritten Mixed Kannada Numerals Recognition Using SVM

G. G. Rajput, Rajeswari Horakeri, Sidramappa Chandrakant
Department of Computer Science, Gulbarga University, Gulbarga, Karnataka-India

Abstract—**A mixer of printed and handwritten numerals may appear in a single document such as application forms, postal mail, and official documents. The process of identifying of such mixed numerals and sending it to respective OCRs is a complex task. In this paper, we present a novel method for recognition of printed and hand written isolated Kannada numerals using single OCR system. Printed/hand written Kannada numeral is scan converted to binary image and normalized to a size of 40 x 40 pixels. The boundary of the numeral is traced and chain code of the image is determined. These codes are represented in a complex plane and 10 dimensional Fourier descriptors are computed that form the feature vector. The 10 dimensional Fourier descriptors are input to multi-class SVM classifier to recognize the numeral class. The proposed algorithm is experimented on 5000 numeral images consisting of handwritten and printed numerals, each of size 2500. The experiment is carried using five-fold cross validation method and yielded recognition accuracy of 97.76%.**

Keywords- printed numerals; handwritten numeral, chain code; Fourier descriptors; SVM

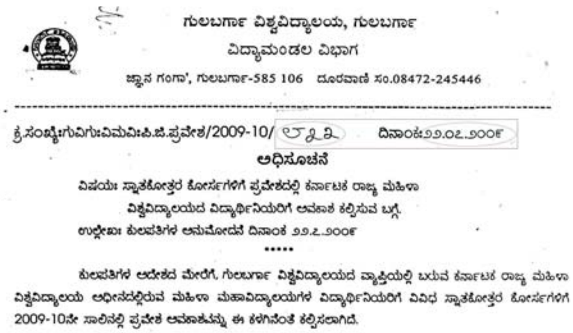
I. INTRODUCTION

The recognition of machine printed and handwritten numerals has been the subject of much attention in pattern recognition because of its number of applications such as bank check processing, interpretation of ID numbers, vehicle registration numbers and pin codes for mail sorting. The performance of character/digit recognition mainly depends on the feature extraction method and the classifier used for labeling the digits. For feature extraction of character recognition, various approaches have been proposed in [1]. An excellent review on different kinds of features and classifiers used for digit recognition is reported in [2]. The features include chain code feature, gradient feature, profile structure feature, and peripheral direction contributivity. The classifiers include the k-nearest neighbor classifier, three neural classifiers, a learning vector quantization classifier, a discriminative learning quadratic discriminant function (DLQDF) classifier, and two support vector classifiers (SVCs). A review on applications of character recognition techniques, methodologies in character recognition, research work in character recognition and some practical OCRs is reported in [3].

The task of classification is to partition the feature space into regions corresponding to source classes or assign class confidences to each location in the feature space. Statistical techniques, neural networks, and more recently support vector machine (SVM) have been widely used for classification due to the implementation efficiency [4-7].

Among studies on Indian scripts, most of the pieces of existing work are concerned about Devanagari and Bangla script characters and digits. Some studies are reported on the recognition of other languages like Telugu, Malayalam, and Kannada. Structural and topological feature based tree classifier and neural network classifiers are mainly used for the recognition of Indian scripts [8]. An overview of OCR research in Indian scripts is reported in [9]. Hanmandlu M, M.Hafizuddin, M Yusuf and V K Madasu [10] have proposed a fuzzy based approach to recognition of multi-font numerals. The preprocessed numeral image is partitioned in to fixed number of sub images called boxes and normalized vector distance of foreground pixels are computed and used as features. Multi-font numeral recognition without thinning based on directional density of pixels is reported in [11]. The outer densities of pixels for each of the direction are computed in four directions viz. bottom, top, left and right. The ratios of these densities are taken with the total area of the cropped numeral image and are used as features. Dinesh Acharya U, N V Subbareddy and Krishnamoorthy [12] have used 10-segment string concept, water reservoir, horizontal and vertical strokes, and end points as features and k-means to classify the Kannada handwritten numerals. Using Image fusion method recognition of Kannada handwritten numerals is reported in [13]. In this method, 64 dimensional features, represented as 8x8 pattern matrices, are used to classify the handwritten Kannada numerals using nearest neighbor classifier. U. Pal, T. Wakabayashi, N. Sharma and F. Kimura [14] have proposed a modified quadratic classifier based scheme towards the recognition of off-line handwritten numerals of six popular Indian scripts. The features used in the classifier are obtained from the directional information of contour points of the numerals. Rajput and Mali [15] have proposed an efficient method for recognition of isolated Devanagari handwritten numerals based on Fourier descriptors. The 64 dimensional Fourier descriptors invariant to translation, scaling, and rotation are fed to SVM classifier for recognition.

In multilingual country like India, it is common that many documents consist of both printed and handwritten characters



ಗುಲಬರ್ಗಾ ವಿಶ್ವವಿದ್ಯಾಲಯ, ಗುಲಬರ್ಗಾ
ವಿದ್ಯಾಮಂಡಲ ವಿಭಾಗ
ಜ್ಞಾನ ಗಂಗಾ', ಗುಲಬರ್ಗಾ-585 106 ದೂರವಾಣಿ ಸಂ.08472-245446

ಕ್ರ.ಸಂಖ್ಯೆ:ಗುವಿಗು:ವಿಮವಿ:ಪಿ.ಜಿ.ಪ್ರವೇಶ/2009-10/ ೮೭೨೩ ದಿನಾಂಕ:೨೨.೦೭.೨೦೦೯

ಅಧಿಸೂಚನೆ

ವಿಷಯ: ಸ್ನಾತಕೋತ್ತರ ಕೋರ್ಸಗಳಿಗೆ ಪ್ರವೇಶದಲ್ಲಿ ಕರ್ನಾಟಕ ರಾಜ್ಯ ಮಹಿಳಾ
ವಿಶ್ವವಿದ್ಯಾಲಯದ ವಿದ್ಯಾರ್ಥಿನಿಯರಿಗೆ ಅವಕಾಶ ಕಲ್ಪಿಸುವ ಬಗ್ಗೆ.
ಉಲ್ಲೇಖ: ಕುಲಪತಿಗಳ ಅನುಮೋದನೆ ದಿನಾಂಕ ೨೨.೭.೨೦೦೯

*****

ಕುಲಪತಿಗಳ ಆದೇಶದ ಮೇರೆಗೆ, ಗುಲಬರ್ಗಾ ವಿಶ್ವವಿದ್ಯಾಲಯದ ವ್ಯಾಪ್ತಿಯಲ್ಲಿ ಬರುವ ಕರ್ನಾಟಕ ರಾಜ್ಯ ಮಹಿಳಾ ವಿಶ್ವವಿದ್ಯಾಲಯ ಅಧೀನದಲ್ಲಿರುವ ಮಹಿಳಾ ಮಹಾವಿದ್ಯಾಲಯಗಳ ವಿದ್ಯಾರ್ಥಿನಿಯರಿಗೆ ವಿವಿಧ ಸ್ನಾತಕೋತ್ತರ ಕೋರ್ಸಗಳಿಗೆ 2009-10ನೇ ಸಾಲಿನಲ್ಲಿ ಪ್ರವೇಶ ಅವಕಾಶವನ್ನು ಈ ಕೆಳಗಿನಂತೆ ಕಲ್ಪಿಸಲಾಗಿದೆ.

Figure 1. Sample document containing handwritten and printed numerals in Kannada

and numerals. Mixture of printed and handwritten character and numerals in Indian context usually appears in a single document such as applications forms, postal mail, office letters, etc. Example is given in figure 1. To the best of our knowledge, there has been very little work related to recognition of both printed and handwritten numerals in Indian scripts [16]. This motivated us to work towards recognition of handwritten and printed mixed numerals. Chain code of the numeral image is computed and the first 10 Fourier descriptors are obtained from the chain code. SVM classifier is used for recognition of test numerals. To validate the proposed method, we have chosen numerals of Kannada script, one of the major scripts in South India. The rest of the paper is described as follows. Description of the proposed method is presented in section 3. Experimental results are discussed in section 3 and conclusion is given in section 4.

II. PROPOSED METHOD

A. Data Collection and Preprocessing

The Kannada language is one of the four major south Indian languages. The Kannada alphabet consists of 16 vowels and 36 consonants. Vowels and consonants are combined to form composite letters. Writing style in the script is from left to right. Further, there are about as many stress marks as there are base characters. Stress marks (vothus) modify the base characters and are appendages. The script also includes 10 different symbols representing the ten numerals of the decimal number system. Fig. 1 presents a listing of the symbols used in Kannada script for the numbers from zero to nine. A brief description of data collection is given below.

Printed numerals come in multi-font styles and sizes. To create the database of printed numerals, multi-font style and multi size numerals were chosen from Nudi 3.0 and Baraha 8.0 Kannada software packages. Numerals of 10 different font-styles namely BRH kasturi, BRH vijaya, Nudi Akshara-01, Nudi Akshara-02, Nudi Akshara-03, Nudi Akshara-04, Nudi Akshara-05, Nudi Akshara-07, Nudi Akshara-09, Nudi B Akshara and BRH Amerikannada of 10 different font sizes 14, 16, 18, 20, 22, 24, 26, 28, 36, 48, and 72 were collected. Few numeral images from printed Kannada documents were also added. A total of 2500 Kannada machine printed numerals were obtained and stored as data set. Sample images of printed numerals are shown in Fig. 3.

0	1	2	3	4	5	6	7	8	9
೦	೧	೨	೩	೪	೫	೬	೭	೮	೯

Figure 2. Kannada numerals 0 to 9

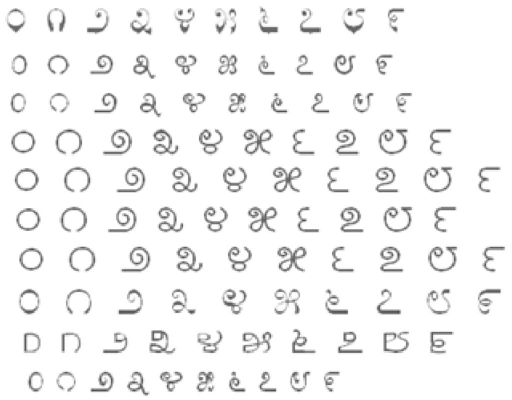


Figure 3. Printed Kannada numerals 0 to 9 of different font style and size

Handwritten numerals usually come in various sizes, shapes and fonts. The database of totally unconstrained handwritten Kannada numerals has been created. Writers were chosen from schools, colleges and professionals and the purpose of numeral collection was not disclosed to them. The collected documents were scanned using HP flatbed scanner which yield low noise good quality gray scale images. It is ensured that the skew introduced during the document scanning is negligible and hence ignored. A total of 2500 handwritten Kannada numerals were obtained and stored as data set. A sample image of scanned document is shown in figure 4.

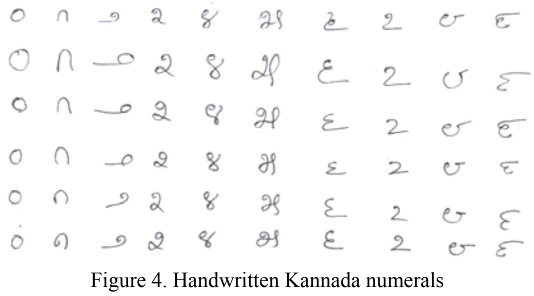


Figure 4. Handwritten Kannada numerals

B. Preprocessing

The first step in preprocessing is to binarize the numeral images (printed and hand written) so that the numeral images have pixel values 0 and 1. A thresholding application has to be performed on scanned gray scale images. Otsu's method [17] has been used for the purpose of selecting the threshold and binarizing the gray scale images, so that resulting image has 0 as background pixels and 1 as foreground pixels. The noise in the image is removed using morphological erode and dilate operations. Further, the spike effect along the boundary of the numerals, due to thrersholding operation, is eliminated using spurring operation [17]. A bounding box is then fitted over the numeral and the numeral is extracted. To bring uniformity among all the numerals, the numerals are normalized to a window size of 40x40 pixels. The 40x40 size of the window is selected due to the fact that the handwritten and printed numerals collected of various sizes will fall around this size. A total of 5000 numeral images representing Kannada machine printed and handwritten numerals, 0 to 9, respectively, are obtained and stored as data set. Each numeral image represents a numeral (binary 1) that is unconstrained, isolated and clearly discriminated from the background (binary 0). The block diagram of preprocessing step is shown in the Fig. 5.

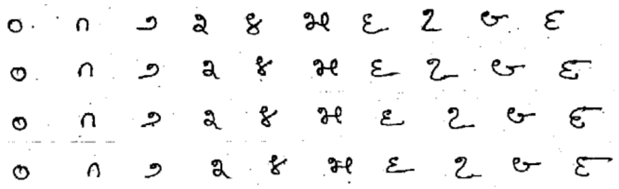


Figure 4. Handwritten Kannada numerals 0 to 9

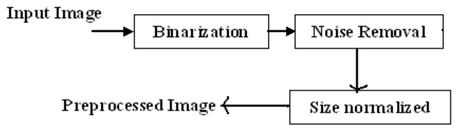


Figure 5. Block diagram of Preprocessing

C. Feature Extraction

Descriptors are some set of numbers that are produced to describe a given shape. The shape may not be entirely reconstructable from the descriptors, but the descriptors for different shapes should be different enough that the shapes can be discriminated. We have used Fourier descriptors, computed from the chain code of the numeral boundary, as features to identify the class label of the numerals. A brief description of these shape descriptors is given below.

1) Chain code: One of the ways to encode the contour or boundary by a connected sequence of straight line segments of specified length and direction is Chain code [17]. The chain code representation is based upon the work of Freeman. We follow the contour in a clockwise manner and keep track of the directions as we go from one contour pixel to the next. The codes associated with eight possible directions are the chain codes and, with x as the current contour pixel position, the codes are generally defined as:

$$\textbf{\textit{Chain Codes}} = \begin{array}{ccc} 3 & 2 & 1 \\ 4 & x & 0 \\ 5 & 6 & 7 \end{array}$$

The algorithm for generating the chain code is given below.

Algorithm 1.

1. Find the boundary of pre-processed numeral image
2. Search the image from top left until a first pixel belonging to the region is found.
3. Search the neighborhood of the current pixel for another pixel of the boundary in clockwise direction
4. The detected inner border is represented by direction code.

2) Fourier Descriptors: Fourier transformation is widely used for shape analysis [18, 19] The Fourier transformed coefficients form the Fourier descriptors of the shape. These descriptors represent the shape in a frequency domain. The lower frequency descriptors contain information about the general features of the shape, and the higher frequency descriptors contain information about finer details of the shape. Although the number of coefficients generated from the transform is usually large, a subset of the coefficients is enough to capture the overall features of the shape. The high frequency information that describes the small details of the shape is not so helpful in shape discrimination, and therefore, they can be ignored. As the result, the dimensions of the Fourier descriptors used for capturing shapes are significantly reduced. The method of computing the transform is explained below.

Suppose that the boundary of a particular shape has K pixels numbered from 0 to $K - 1$. The k-th pixel along the contour has position (x_k, y_k). Therefore, we can describe the contour as two parametric equations:

$$x(k) = x_k$$

$$y(k) = y_k$$

We consider the $(x,\ y)$ coordinates of the point not as Cartesian coordinates but as those in the complex plane by writing

$$s(k) = x(k) + i\,y(k)$$

We take the discrete Fourier Transform of this function to end up with frequency spectra.

The discrete Fourier transform of s(k) is

$$a(u) = \frac{1}{k}\sum_{k=0}^{K-1} s(k)e^{-j2\pi uk/K}, \quad u = 0, 1, …., K\text{-}1 \qquad (1)$$

The complex coefficients a(u) are called the *Fourier descriptors* of the boundary. The inverse Fourier transform of these coefficients restores s(k). That is,

$$s(k) = \sum_{u=0}^{K-1} a(u)\,e^{j2\pi uk/K}\ , \quad k = 0, 1, …., K\text{-}1$$

The algorithm for computing the Fourier descriptors is given below.

Algorithm 1.

1. Represent the codes, obtained by performing algorithm 1 on the numeral image, in the complex plane using the lookup table 1.
2. Apply Fourier transform to these set of numbers using equation (1) and obtain first ten Fourier descriptors.
3. Store these 10 dimensional descriptors as feature vector of the numeral.

The number of descriptors, defining the numeral, are chosen to be 10, by performing experiments to determine the number of descriptors to be retained enough to classify the numeral.

Table 1. Lookup table

Chain code	0	1	2	3	4	5	6	7
Complex number	1	1+i	i	-1+i	-1	-1-i	- i	1-i

3) Pattern classifier and recognition: Support vector machine (SVM) is defined for two-class problem and it finds the optimal hyper-plane which maximizes the distance, the margin, between the nearest examples of both classes, named support vectors (SVs). Given a training database of M data: {xm| m=1,...,M}, the linear SVM classifier is then defined as:

$$F(x) = \sum \alpha_j \, x_j \, . \, x + b$$

where {xj} are the set of support vectors and the parameters αj and b have been determined by solving a quadratic problem. The linear SVM can be extended to a non-linear classifier by replacing the inner product between the input vector x and the SVs xj, through a kernel function K defined as:

$$K(x, y) = \phi\,(x) \, . \, \phi(\, y)$$

This kernel function should satisfy the Mercer's Condition [20]. The performance of SVM depends on the kernel. Commonly used kernels include linear Kernel, Radial Basis Function (Gaussian) Kernel, and Polynomial Kernel. We have used RBF (Gaussian) kernel, which outperformed the other commonly used kernels in the preliminary experiments.

Properties of SVMs:

Complexity of training: SVMs are trained by quadratic programming (QP), and the training time is generally proportional to the square of number of samples. Some fast SVM training algorithms with nearly linear complexity are available however.

Flexibility of training: SVMs can be trained at the level of holistic patterns.

Model selection: The QP learning of SVMs guarantees finding the global optimum. The performance of SVMs depends on the selection of kernel type and kernel parameters, but this dependence is less influential.

Classification accuracy: SVMs have been demonstrated superior classification accuracies in many experiments.

Storage and execution complexity: SVM learning by QP often results in a large number of SVs, which should be stored and computed in classification.

An excellent tutorial on SVM is given in [21].

III. EXPERIMENTAL RESULTS

Presented in this section are the recognition accuracies obtained using normalized binary images representing Kannada numerals. The proposed system is experimented on the following dataset.

1) 2500 handwritten Kannada numerals

2) 2500 printed Kannada numerals

3) 5000 printed and handwritten mixed Kannada numerals

For result computation k-fold cross validation technique was adopted. When using the k-fold method, the dataset is randomly partitioned into k groups. The SVM algorithm is then trained k times, using all of the training set data points except those in the kth group. The form of the algorithm is as follows:

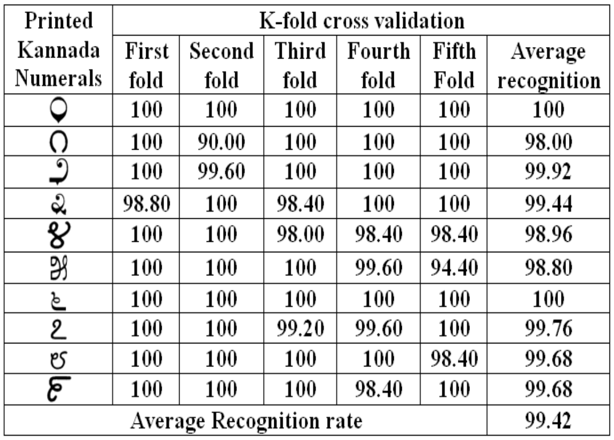
- Divide the data set into k partitions.
- For each k:
    - Make T the dataset that contains all training data points except those in the kth group.
    - Train the algorithm using T as the training set.
    - Test the trained algorithm, using the kth set as the test set.

For the proposed method we have chosen k=5. The proposed method is implemented using Matlab 6.1 software using Matlab Tool Box for Pattern Recognition(PRTools[22]).

Table 1 presents the results for printed Kannada numerals and Table 2 presents the results for handwritten Kannda numerals. The recognition rate obtained for printed numerals is 99.42% and for handwritten numerals is 97.34%. Clearly, the results obtained for printed numerals are better compared to the results of handwritten numerals. This is because of large variation in handwritten numerals. Further, feature vector for certain numerals (for eg. numerals 3 and 7) of different class are found to be similar and hence misrecognized.

Table 3 presents the results for handwritten and printed mixed Kannada numerals. The 5- fold cross validation has yielded an overall recognition rate of 97.76% which is higher compared to the results of handwritten numerals (Table2). The recognition rate can certainly be enhanced by performing experiments on a large dataset.

Table 2: Recognitions results of Printed Kannada numerals



Printed Kannada Numerals	K-fold cross validation					
	First fold	Second fold	Third fold	Fourth fold	Fifth Fold	Average recognition
೦	100	100	100	100	100	100
೧	100	90.00	100	100	100	98.00
೨	100	99.60	100	100	100	99.92
೩	98.80	100	98.40	100	100	99.44
೪	100	100	98.00	98.40	98.40	98.96
೫	100	100	100	99.60	94.40	98.80
೬	100	100	100	100	100	100
೭	100	100	99.20	99.60	100	99.76
೮	100	100	100	100	98.40	99.68
೯	100	100	100	98.40	100	99.68
Average Recognition rate						99.42

Table 3: Recognition results of handwritten Kannada numerals

Handwritten Kannada Numerals	K-fold cross validation					
	First fold	Second fold	Third fold	Fourth fold	Fifth Fold	Average recognition
೦	100	100	100	100	100	100
೧	96.80	100	100	100	100	99.36
೨	97.20	94.00	94.00	90.00	100	95.04
೩	94.00	94.00	93.20	93.20	93.80	93.64
೪	97.60	97.20	96.80	96.80	100	97.68
೫	98.40	97.60	94.00	98.40	98.80	97.44
೬	96.80	98.40	99.60	96.80	98.80	98.08
೭	94.00	93.00	97.20	94.00	97.60	95.16
೮	97.20	97.20	97.20	100	100	98.32
೯	99.60	100	95.00	100	98.80	98.68
Average Recognition rate						97.34

Table 4: Recognition results of printed and handwritten mixed Kannada numerals

Printed & Handwritten Mixed Kannada Numerals	K-fold cross validation					
	First fold	Second fold	Third fold	Fourth fold	Fifth Fold	Average recognition
೦	100	100	100	100	100	100
೧	98.00	97.00	100	100	99.00	98.83
೨	100	98.00	97.00	100	99.00	98.96
೩	96.00	93.00	94.20	96.00	94.20	94.68
೪	94.20	94.20	99.00	94.20	99.00	96.12
೫	95.00	97.00	98.00	97.00	96.00	96.06
೬	98.00	99.00	97.00	99.00	97.20	98.04
೭	97.20	98.00	96.00	97.20	98.00	97.28
೮	100	97.20	100	97.00	99.90	98.92
೯	99.00	99.00	99.00	96.00	100	98.73
Average Recognition rate						97.76

IV. CONCLUSIONS

A document containing both printed and handwritten numerals is common in Indian environment. In such cases, it is essential to have a single OCR for recognition of the numerals, printed or handwritten. In this paper, a novel and efficient method for recognition of printed and handwritten mixed Kannada numerals is presented. Features are obtained by extracting the chain code of the numeral and computing Fourier descriptors from these codes. The features are fed to multi-class SVM for recognition. The proposed method has yielded recognition accuracy of 97.76%. The method has also been tested separately for handwritten and printed numerals, respectively, and has yielded good results. The proposed method can be extended to the numerals of other scripts.

ACKNOWLEDGMENT

We thank the referees for their helpful comments. Our thanks to Dr. P. S. Hiremath and Dr. B. V. Dhandra, Department of Computer Science, Gulbarga University, Gulbarga, for their helpful suggestions.

REFERENCES

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

[2] Liu C. L., Nakashimga, K. Sako, H. Fujisasa, Handwritten Digit Recognition: Benchmarking of the state-of-the-art techniques, Pattern Recognition 36, 2003, pp 2271-2285.

[3] Govindan V K, Shivaprasad A P, Character Recognition – a review, Pattern Recognition 23, 1990, pp 671-683.

[4] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, 2nd Edition, Wiley Interscience, New York, 2000.

[5] A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1), 2000, pp 4–37.

[6] C.M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1995.

[7] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Knowledge Discovery Data Mining 2 (2), 1998, pp 1–43.

[8] U. Pal,B B Chaudhuri, "Indian script character recognition: a survey", pattern Recognition 37, 2004, pp 1887-1899.

[9] B. Anuradha Srinivas, Arun Agarwal and C.Raghavendra Rao, An Overview of Ocr Research in Indian Scripts. International Journal of Computer Sciences and Engineering Systems(IJCSES), vol.2, No.2, 2008, pp 141-150.

[10] Hanmandlu M, M.Hafizuddin, M Yusuf and V K Madasu , Fuzzy based Approach to Recognition of Multifont Numerals , Proc. of 2nd National Conf. on Document Analysis and Recognition (NCDAR), Mandya, 2003, pp 118-126.

[11] Dhandra, B.V, Malemath V.S, Mallikarjun H, Hegadi R, Multi-font Numeral Recognition without Thinning based on Directional Density of Pixels, IEEE, vol 1 2006, pp 157-160.

[12] Dinesh Acharya U, N V Subbareddy and Krishnamoorthy, Multilevel Classifier in Recogniton of Handwritten Kannada Numeral, *Proceedings of World Academy of Science*, Engineering And Technology, vol. 32, 2008, pp 308-313.

[13] G. G. Rajput and Mallikarjun Hangarge, Recognition of Isolated Kannada Numeral Based on Image Fusion Method. PReMI 2007, LNCS 4815, 2007, pp. 153–160.

[14] U. Pal, T. Wakabayashi, N. Sharma and F. Kimura, Handwritten Numeral Recognition of Six Popular Indian Scripts. In Proc. 9th *International Conference on Document Analysis and Recognition*., Curitiba, Brazil, September 24-26, 2007, pp. 749-753.

[15] G. G. Rajput and S. M. Mali, Fourier descriptor based Isolated Marathi Handwritten numeral Recognition, International journal of Computer Application, 10.pp 5120/724-1017, 2010.

[16] A MLP Classifier for Both Printed and Handwritten Bangla Numeral Recognition, A. Majumdar and B. B. Chaudhuri, Computer Vision, Graphics and Image Processing, LNCS, Springer, Volume 4338/2006, pp 796-804.

[17] Rafael C. Gonzalez, Richard E. Woods, *Digital Image Processing*. Pearson Education Asia, 2nd Edition, 2002.

[18] Eric Persoon and King-sun Fu 1977. Shape Discrimination Using Fourier Descriptors. IEEE Trans. On Systems, Man and Cybernetics, Vol. SMC- 7(3), 1977, pp170-179.

[19] Fethi Smach, Cedric Lemaître, Jean-Paul Gauthier Johel Miteran, Mohamed Atri. Generalized Fourier Descriptors with Applications to Objects Recognition in SVM context, Journal of Mathematical Vision and Imaging,Vol 30(1), 2008, pp 43-71.

[20] V. Vapnik. The Nature of Statistical Learning Theory, *Springer Verlang,* 1995.

[21] C. J. C. Burges, A tutorial on support vector machines for pattern recognition., *Data Mining and Knowledge Discovery*, 1998, pp 121-167.

[22] http://prtools.org/