KANNADA HANDWRITTEN CHARACTER RECOGNITION FOR AUTOMATIC FORM PROCESSING

[1]SHRUTHI M, [2]K.V. SURESH
[1]M.Tech student, [2]Professor
Siddaganga Institute of Technology, Tumakuru, Karnataka, India

Abstract— Data processing and management is common now a days. In this paper, automatic processing of forms written in Kannada language is considered. A suitable pre-processing technique is presented for extracting handwritten characters. Principal Component Analysis (PCA) and Histogram of oriented Gradients (HoG) are used for feature extraction. These features are fed to multilayer feed forward back propagation neural network for classification. Only 57 characters are used for recognition. Performances of two features are compared for different number of classes. HoG is found to have better recognition accuracy than PCA as number of classes increased. This is implemented in Visual Studio 2010 using OpenCV library.

Keywords—Back Propagation Neural Network, Form Processing, Histogram of Gradients, Kannada Script, Principal Component Analysis.

I. INTRODUCTION

The automation of handwritten form processing is attracting intensive research interest due to its wide application and reduction of manual work. In Indian context, many organisations will collect the data on paper based forms. Automatic processing of these forms is a process of capturing the information stored in the forms and converting it into electronic (machine readable) format.

Handwritten character recognition contributes immensely to the advancement of automation process. This is classified into offline and online recognition. For Automatic Form Processing (AFP) offline recognition method is used since it involves automatic conversion of text in an image into letter codes which are usable within computer and text processing applications [1]. AFP includes complete scan of a form using scanner. The scanned image then undergoes various pre-processing operations, character segmentation and recognition of handwritten characters.

India is a multi-lingual and multi-script country comprising of eighteen official languages, Kannada is one among them. Several works has been done for the recognition of handwritten Kannada characters. The major pre processing steps of AFP includes edge detection, morphological operations to make it suitable for segmentation. Segmentation separates the image text documents into lines, words and the characters. Thungamani M and RamakanthKumar P [2] discuss two segmentation techniques such as classical approach and holistic approach. In classical approach, the input image is segmented into sub images. In holistic approach, the characters are recognized without dissection. Mamatha H.R and Srikanatamurthy K [3], proposed a segmentation scheme using projection profiles. Morphological operations are used to remove the noise. After this text lines are extracted using horizontal projection profile, words and characters are extracted using vertical projection profiles.

Kannada script has large number of character set. This may reduce the recognition accuracy and increase the computational cost. To avoid this problem an algorithm has been proposed to reduce symbol set [4], where the vowel modifiers (kagunitha) and consonant modifiers (vattakshara) which are not connected to base characters are considered as separate classes.

Devanagari script has similar characteristics as Kannada script like vowel modifiers, consonant conjuncts etc,. The recognition of this script consists of three phases: segmentation, decomposition, i.e,. decomposing a composite character into base part and modifier parts; and recognition [5]. Only small subsets of compound characters (upper and lower signs) are considered for the recognition. Many Arabic letters also share common primary shapes, which differs only in the number of dots and the dots or above or below the primary shape. A survey on offline recognition of Arabic handwriting recognition is presented in [6]. Different segmentation, feature extraction and recognition engines used for OCR are also discussed.

An overview of character recognition methodologies with respect to the offline character recognition systems such as pre-processing, segmentation, representation, recognition and post-processing methods are presented in [7]. Shape based features such as Fourier descriptors and chain codes are used for the recognition of handwritten Kannada characters (vowels and numerals) are discussed in [8]. Support Vector Machine (SVM) is used for recognition purpose and an accuracy of 95% is obtained. A brief survey on offline recognition of Devanagari script is presented in [9]. Performance of different feature extraction techniques using different classifiers is tabulated. Gradient and PCA based features with PCA, SVM and Neural Network classifiers are found to have better recognition accuracy.

Development of two databases for two popular Indian scripts Devanagari and Bangla for numeral recognition is presented in [10]. This uses a multistage cascade recognition scheme using wavelet-based multi-resolution representations and multi layer perceptron (MLP) classifiers to achieve higher recognition accuracy. This is then used to the recognition of mixed numerals for three Indian scripts such as Devanagari, Bangla and English.

Unconstrained handwritten recognition of Kannada characters using very large dataset of 200 samples using ridgelet transform is discussed in [11]. To reduce the dimension of feature vector PCA is used. It is found that ridgelet features offered promising result than PCA. A zone based method for the recognition of handwritten Kannada vowels and consonants is presented in [12]. Character image is divided into 64 non-overlapped zones and from each zone crack codes are computed. SVM is used as classifier and an accuracy of 87.24% is achieved.

Literature records few papers on Kannada character recognition. Choice of methods for feature extraction is important for achieving efficient character recognition for large classes. In this paper, Kannada handwriting recognition for automatic form processing is considered. PCA and HoG are used for feature extraction. Performances of features are compared for 57 classes.

II. KANNADA SCRIPT

Kannada alphabets were developed from the descendents of Brahmi script such as Kadamba and Chalukya scripts. Handwritten character recognition of Kannada characters is a very challenging task because of its large dataset, shape similarity among characters and non-uniqueness in the representation of diacritics. Kannada script has 49 primary characters: 15 vowels and 34 consonants as shown in Fig. 1(a) and Fig. 1(b). Each of the vowels, modify primary consonants to form a compound character or kagunita as shown in Fig. 1(c). Additionally a consonants emphasis glyph called consonant conjuncts shown in Fig. 1(d) [vattakshara], exists for each of the 34 consonant. Thus, total number of possible combinations is as shown in TABLE 1.

C D E F G H IÄ J K L M N O CA CB
(a) Vowels

PÀ R UÀ WÀ Y
ZÀ bÀ d gÀhÄ k
l oÀ qÀ qsÀ t
vÀ xÀ zÀ zsÀ £À
¥À ¥sÀ § ¨sÀ ªÀÄ
AiÀÄ gÀ ® ªÀ ±À µÀ ¸À ºÀ ¼À

(b) Consonants

PÀ PÁ Q QÃ PÀÄ PÀÆ PÀÈ PÉ PÉÃ PÉÊ PÉÆ PÉÆÃ PË PÀA PÀB

(c) Vowel modified consonants

Ì Í Î Ï Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ ß à á â ã ä å ç è é ê ë ì í î

(d) Consonant conjuncts

Fig. 1: Basic Kannada character set

Table I

Maximum possible combinations of Kannada characters

Character Type	V (Vowel)	C (Consonant)	CV (Kagunita)	CCV (tÆÚ)	CCCV (vÀì÷å)	N	Total
Possible combinations	15	34	510	17340	589560	10	589585

The problem of large number of dataset can be reduced by considering each of the non connected characters as different symbols.

III. FORM PROCESSING METHODOLOGY

Automatic Form Processing system involves Image acquisition using scanner, pre-processing of scanned form, only handwritten character extraction, handwritten character segmentation, character recognition and storage as shown in Fig. 2.

The template of the Birth certificate is created with all the required data fields. The applicant is then instructed to fill the form in Kannada language with all the base characters in the upper box and conjuncts in the lower box. The filled form is then scanned using flatbed scanner for processing.

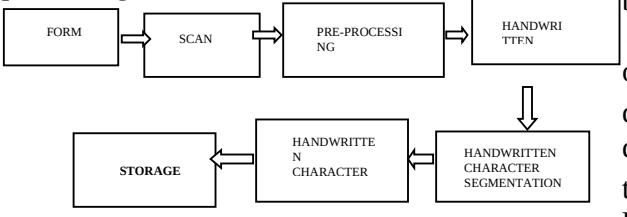

Fig. 2: Block diagram of Automatic Form Process

The form is then pre-processed to remove the effects of scanning such as skew detection and correction, noise and the blur. Template of the form consists of printed characters, handwritten characters and bounding boxes. Only the handwritten characters are extracted using morphological subtraction. Extracted characters are then recognised using suitable recognition engine.

A. Pre-processing

The scanned image is pre-processed to eliminate the imperfections and to remove the uninformative variations in handwriting. The major pre-processing steps include resize, binarization, edge detection, skew detection and correction, dilation and erosion as shown in Fig. 3.

Each scanned image is resized to 1024 x 1024 standard dimensions. The resized gray scale image is converted into binary. Edges are detected using Canny edge detector, since it gives all the possible edges present in the image. Skew angle of scanned image is detectedusing Hough transform

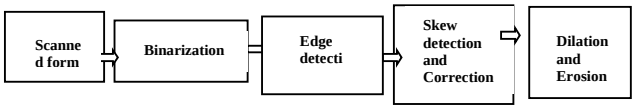

Fig. 3: Pre-processing steps of form processing

The simplest case of the Hough transform is the linear transformation for detecting straight lines. In the image space, a straight line can be represented as $y = mx + b$ and graphically plotted for each pair of image points (x, y) . The characteristics of the straight line are described in terms of slope m and intercept b . But, vertical lines leads to unbounded values of m and b . Thus, the straight line is represented using polar parameters r and θ as:

$$r = x\cos\theta + y\sin\theta \quad (1)$$

The parameter r represents the distance between the straight line and the origin and the θ represents the angle of the vector from the origin to its closest point. Each point in the (x, y) domain is mapped onto the curves in the domain (r, θ) . Thus the equation (1) represents the family of lines passing through particular line (x, y) . If the curves of two different points intersect each other in polar space that indicates that both the points belong to same line in image space. The angle at the point of intersection gives the angle of straight line as shown in Fig. 4(b).

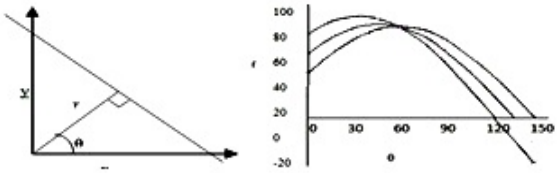

Figure 4: (a) Mapping of x and y to polar space. (b) Curves in polar space.

After finding the skew angle the document is skew corrected using rotational affine transformation which uses cubic interpolation technique. Morphological operations are performed to remove the printed data present in the document. Dilation adds pixels to the boundaries of objects in an image using a 3 x 3 structuring element to find the local maxima and an output matrix is created from these maximum values as shown in equation (2). Erosion removes the pixels on object boundaries by finding its local minima and creates an output matrix as shown in equation (3).

$$A \oplus B = \left\{ z \mid \left(\hat{B}\right)_z \cap A \neq \Phi \right\} \quad (2)$$

$$A \Theta B = \left\{ z \mid \left(B\right)_z \subseteq A \right\} \quad (3)$$

The handwritten characters from the data entry fields are obtained by filling the region with holes.

After pre processing an image with sequence of characters is segmented into individual character using connected component analysis. For each connected component a rectangular bounding box is fitted. By using the properties of region and bounding box, each character is cropped and stored.

B. Feature Extraction

Features are a set of numbers that capture the salient information of the segmented characters. In this work Principal Component Analysis (PCA) without dimensionality reduction and HoG are used for feature extraction.

PCA transformation converts the correlated variables into uncorrelated variables retaining the maximum amount of variations. Each image is converted to single unit vector and stored as columns of large matrix $P X N$, where N is the number of images and P is the vector dimension. Mathematically principal components for $P X N$ matrix $[X_1, X_2, \ldots X_N]$ are obtained by subtracting mean from each column. The resultant matrix is given by $B = [\hat{X}_1, \hat{X}_2, \ldots \hat{X}_N]$. Then the covariance matrix S is obtained by $S = \frac{1}{N-1} BB^T$. The eigen values and eigen vectors are calculated using $P^T SP = D$ where D is a diagonal matrix with the eigen values and P is a matrix of eigen vectors. Eigen vectors are arranged according to the descending values of the eigen values. The weight matrix W is determined as $W = PB^T$. This weight matrix is of dimension $P X N$ is used for classification of characters.

Histogram of Oriented Gradient (HOG) is a feature extraction technique used in computer vision and image processing applications for the purpose of object detection. This technique counts the occurrences of gradients orientation in localized portions of the image. The implementation of this technique can be achieved by dividing image into small connected regions, called cells. For each cell magnitude of the gradient and directions are computed. Then the histogram of gradients is computed using 9 channels. This histogram is used as feature. The magnitude of the gradient and directions are given by $|G| = \sqrt{I_x^{\,2} + I_y^{\,2}}$ and $\theta = \arctan \frac{I_y}{I_x}$.

C. Handwritten Character Classification

The classification stage is the decision making part of a recognition system and it uses the features extracted in the previous stage. The classification capability of the network depends on its architecture and learning rule. The architecture considered here is feed forward back propagation neural network with MLP. All neurons in the architecture use sigmoid output function.

IV. EXPERIMENTAL RESULTS

For automatic form processing, a template of the birth certificate is created with all the relevant fields. The applicant is instructed to fill the form in Kannada language with base letters in the upper box and conjuncts in the lower box as shown in Fig. 5. The filled form is then scanned to extract the written information. Skew corrected image is shown in Fig. 6. An example, showing Kannada form processing for 57 characters is shown in Fig. 7. The result of Dilation is shown in Fig. 8. The results of data field identification and erosion are shown in Fig. 9 and Fig. 10. Eroded image multiplied with the original image and hole filled to it is shown in

Fig. 11 and Fig. 12. The result of segmentation of handwritten characters by fitting bounding boxes is shown in Fig. 13.

The segmented characters are used for feature extraction using PCA and HoG. Performance of these feature for different number of classes are compared using neural network with back propagation learning. 100 samples per class are used for recognition purpose. Neural network parameters are listed in TABLE 2. Recognition accuracy of PCA and HoG for different number of classes are listed in TABLE 3. The recognised characters are stored in database in relevant fields for future use as shown in Fig. 14.

Table 2: Neural Network Training Parameters

Number of Input nodes	270
No. of Hidden layer	1
No. of Hidden nodes	125
Training epochs	30000
Goal achieved	10e-6
No. of Output nodes	57

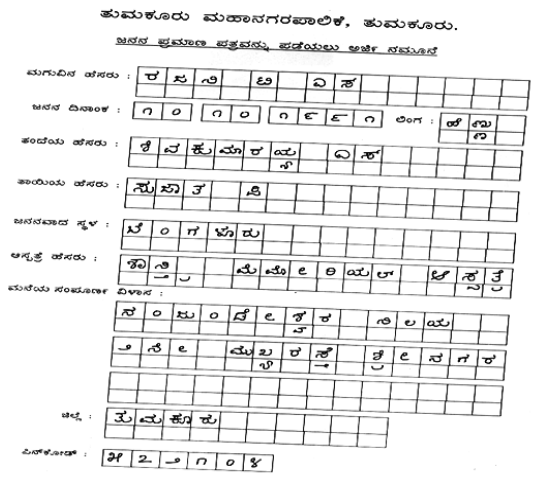

Figure 5: Original skewed form

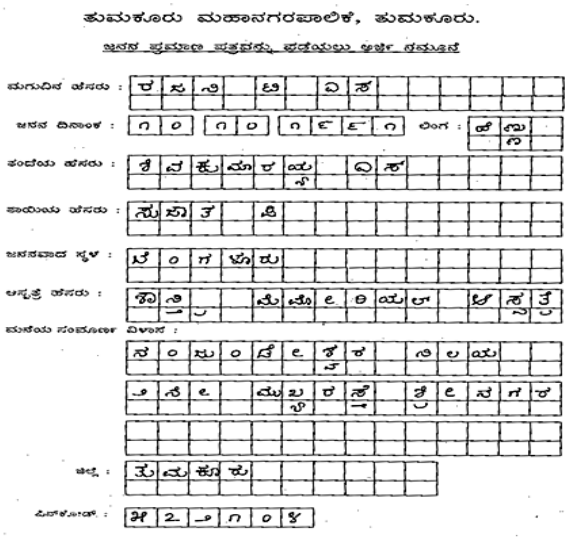

Figure 6: Skew corrected Image

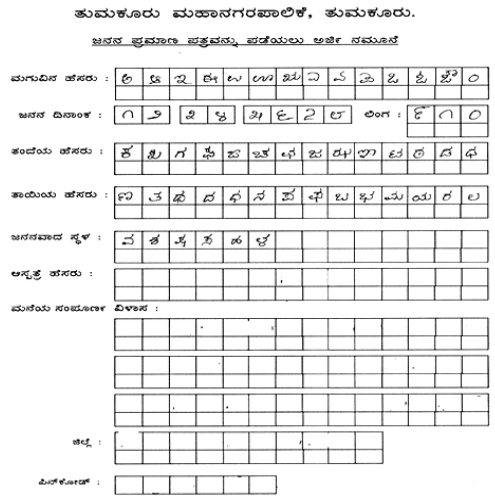

Figure 7: Form with 57 characters

Each image is resized to 25 x 20, thus the input layer has 500 neurons. Here, only the recognition of 57 Kannada characters is considered which includes 13 vowels, 10 numbers and 34 consonants. Thus, the number of output neurons is 57. Initially vowels and numbers are considered separately for classification. 100 samples in each class and one hidden layer is used. Later the number of classes increased to 23 by combining vowels and numerals. Then including consonants the number of classes becomes 57.

PCA represents the image in the direction of its maximum variance, whereas HoG capture the image variability in 9 directions. When the number of classes which have shape similarity among them increases, HoG gives better performance than PCA.

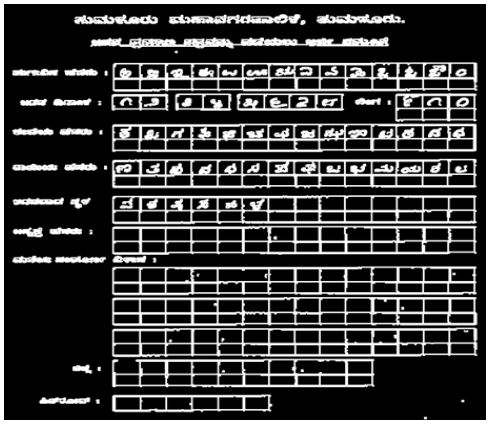

Figure 8: Dilated form

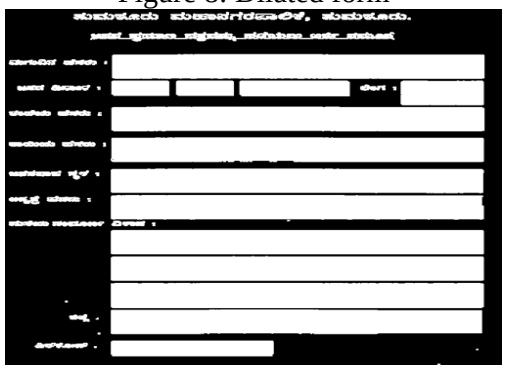

Figure 9: Data fields Identification

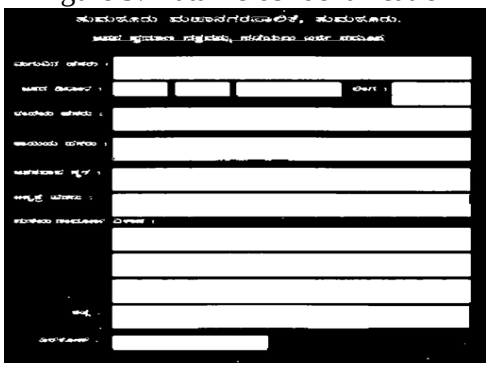

Figure 10: Eroded Image

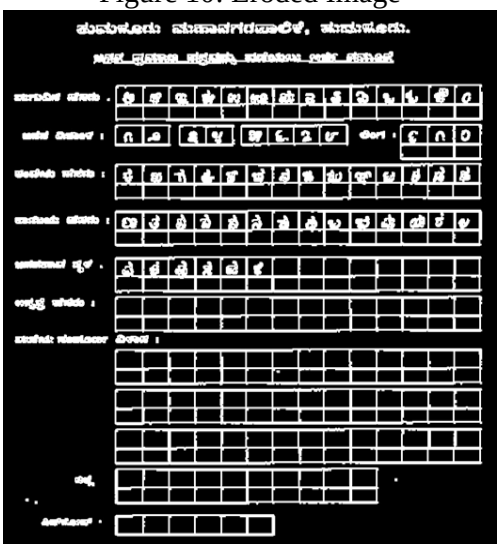

Figure 11: Eroded image multiplied with original

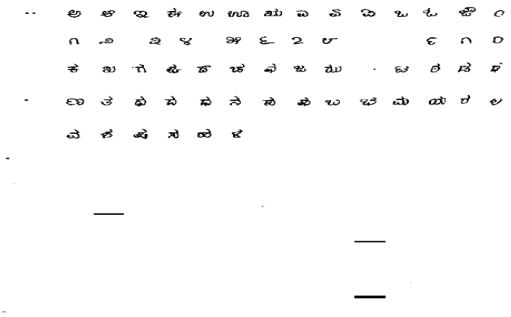

Figure 12: Hole filled Image

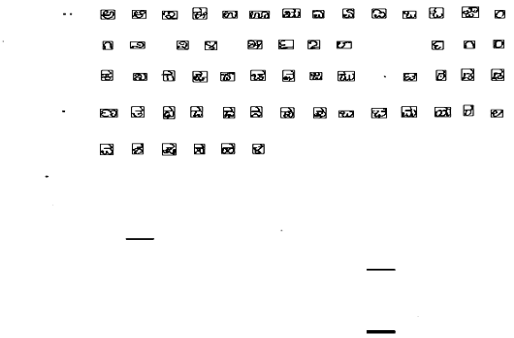

Figure 13: Segmented characters

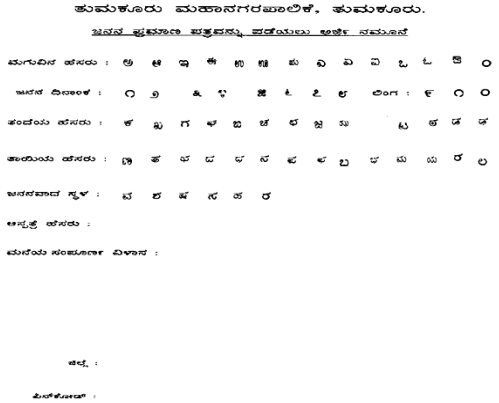

Figure 14: Recognised characters stored in database

Table 3: Recognition accuracies of PCA and HoG

No. of classes	PCA(%)	HOG (%)
Numbers (10)	99.3	99.6
Vowels (13)	97	98
Numbers+Vowels(23)	71	92.07
Consonants(34)	68	78.9
Basic characters (57)	59.35	84.62

V. CONCLUSION

In this paper, Kannada character recognition with application to automatic form processing is presented. Only 57 characters are considered for processing. Using suitable pre-processing techniques only handwritten characters are extracted. PCA and HoG are used for feature extraction. Performance of features is compared using neural network classifier. Results of pre-processing implemented using Visual Studio using OpenCV library are presented. HoG is found to have better recognition accuracy than PCA for large number of classes.

REFERENCES

[1] Afef Kacem, Asma Saidani, Abdel Belaid, "A system for an automatic reading of student information sheets", International Conference on Document Analysis and Recognition, pp 1265-1269, 2011.

[2] M. Thungamani and P. Ramakanth Kumar, "A Survey Methods and Strategies in Handwritten Kannada Character Segmentation", International Journal of Science Research, Vol 1, Issue 1, June 2012.

[3] H.R Mamatha and and K. Srikantamurthy, "Morphological operations and projection profile based segmentation of handwritten Kannada document", International Journal of Applied Information Systems (IJAIS), Vol 4, No.5, October 2012.

[4] Nethravathi B, Archana C.P, Shashikiran K, A.G Ramakrishnan and VIjay Kumar, "Creation of huge annotated database for Tamil and Kannada OHR", 12th IEEE International Conference on Frontiers in Handwriting Recognition, Nov 2010, Pages 415-420.

[5] R M. K. Sinha and H. N. Mahabala, "Machine Recognition of Devanagai Script", IEEE Transactions on Systems, Man and Cybernetics, Vol. Smc 9, No 8, August 1979.

[6] Liana M. Lorigo and Venu Govindaraju, "Offline Arabic handwriting recognition: A survey", IEEE Transactions on Pattern Analysis and machine Intelligence, Vol 28, No.5, May 2006.

[7] Nafiz Arica and Fatos T. Yarman-Vural, "An overview of character recognition focused on offline handwriting", IEEE Transactions on Systems, Man and Cybernetics-Part C: Application and Reviews, Vol 31, No. 2, May 2001.

[8] Rajput G.G and Horakeri R, "Shape descriptors based handwritten character recognition engine with application to Kannada characters", IEEE International Conference on Computer and Communication Technology, Sep 2011, Pages 135-141.

[9] R. Jayadevan, Satish R Kolhe, Pradeep M Patil and Umapada Pal, "Offline recognition of Devanagari script: A survey", IEEE Transactions on Systems, Man and Cybernetics –Part C: Applications and Reviews, Vol 41, No. 6, November 2011.

[10] Ujjwal Bhattacharya, and B.B. Chaudhuri, "Handwritten Numeral Databases of Indian Scripts and Multistage Recognition of Mixed Numerals", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 31, No. 3, March 2009.

[11] C. Naveena and V.N. Manjunath Aradhya, "An Impact of Ridgelet Transform in Handwritten Recognition: A Study on Very Large Dataset of Kannada Script", IEEE 2011 World Congress on Information and Communication Technologies, Pages: 618-621, December 2011.

[12] Ganpat Singh G Rajput and Rajeshwari Horakeri, "Zone based Handwritten Kannada Character Recognition Using Crack code and SVM", International Conference on Advances in Computing, Communications and Informatics (ICACCI), Publisher: IEEE, Aug 2013, Pages 1817-1821.