Off-line Handwritten Kannada Text Recognition using Support Vector Machine using Zernike Moments

Thungamani.M[1] **Dr Ramakhanth Kumar P**[2] **Keshava Prasanna**[3] **Shravani Krishna Rau**[4]

[1,3] Research Assistant, Tumkur University, Tumkur
[2] Professor and HOD, R.V. College of Engineering, Bangalore
[4] Student, R.V.College of Engineering

Abstract
It is a well-known fact that building a character recognition system is one of the hottest areas of research as it is shown over the Internet and due to its wide range of prospects. The objective of this paper is to describe an OCR system for handwritten text documents in Kannada. The input to the system is a scanned image of a text and the output is a machine editable file compatible with most typesetting Kannada software. The system first extracts characters from the document image and a set of features are extracted from the character image using Zernike moments. The final recognition is achieved using support vector machine (SVM). The recognition is independent of the size of the handwritten text and the system is seen to deliver reasonable performance.

1. Introduction

The objective of Document Image Analysis is to process the image of a hand-written text page and render the information contained there into a form suitable for easy modification and manipulation on a computer. In general, document image analysis consists of two parts: textual processing and graphical processing.
Graphical processing is intended to handle the graphical parts such as figures, photographs etc. in the handwritten document while textual processing is for handling the text contained in the document.
A document image analysis system is one that can handle handwritten text documents in Kannada, which is the official language of the south Indian state of Karnataka. The input to the system is the scanned image of a page of handwritten Kannada text. The output is an editable computer file containing the information in the handwritten page.
The system is designed to be independent of the size of characters in the handwritten document and hence can be used with any kind of handwritten document in Kannada. Currently there are many OCR systems available for handling handwritten English documents with reasonable levels of accuracy. (Such systems are also available for many European languages as well as some of the Asian languages such as Japanese, Chinese etc.). However, there are not many reported efforts at developing OCR systems for Indian languages. (See, e.g., Sinha and Mahabala 1979, Choudhury and Pal 1997, Bansal and Sinha 1999, Antani and Agnihotri 1999, for OCR systems for some of the Indian languages.)[1, 2]. The work reported in this project is motivated by the fact that there are no reported efforts at developing document analysis systems for the south Indian language of Kannada.

1.1 The Kannada script

The Kannada alphabet is classified into two main categories: vowels and consonants [3]. There are 16 vowels and 35 consonants. Words in Kannada are composed of aksharas, which are analogous to characters in an English word. While vowels and consonants are aksharas, the vast majority of aksharas are composed of combinations of these in a manner similar to most other Indian scripts.

2. System Presentation

In our document analysis system, the sequence of processing steps is as follows The page of text is scanned through a scanner @ 300 DPI and binarized using a global threshold computed automatically based on the specific image.This image is first processed to remove any skew so that the text lines are aligned horizontally in the image. A set of 56 features are extracted from each character image and these features are sent to the classifier as inputs. The output of the classifier is transliterated so that it can be loaded into any of the typesetting Kannada software for viewing or editing purpose.

3. Pre-processing

Skew estimation and correction of the document image
That is, the angle formed by the horizontal axis of the document image and the text lines. A method based on the Wigner-Ville distribution of the horizontal projection

Manuscript received July 5, 2011
Manuscript revised July 20, 2011

profile of the document image is used. This approach is capable of localizing the areas of the handwritten document images with different skew angles automatically and handling each of them separately. Moreover, it is fast and accurate.

Slope and slant correction.

The term slope of a word is referred to the angle formed by the horizontal axis of the word and the corresponding text line while the term slant of characters is referred to characters whose vertical strokes are inclined regarding the vertical axis of the word. A similar method with the one used for skew estimation is followed. This approach is based on a simple algorithm and achieves very low response time cost. Moreover, it is not dependent on special characters. In the next section a brief presentation of the Wigner-Ville distribution, as it is used in the proposed system, is given several methods have been proposed to solve the problem of skew detection. O'Gorman categorization of these methods according to their techniques into three categories: projection profile, Hough transforms and nearest neighbour clustering. Moreover, hybrid systems using more than one technique have also been proposed.

In this paper we have implemented the generic approach to skew angle estimation, based on the Wigner Ville distribution developed by [4,5,6] E.Kavallieratou et.al.

3.1 Wigner-Ville Distribution

The horizontal and vertical projection profiles of a black and white document image f(x, y) are

$$P_h(x) = \sum_{y} f(x,y)$$

$$P_v(y) = \sum_{x} f(x,y)$$

By introducing the above equations in the WVD of a discrete signal, the space-frequency distribution is expressed as

$$W(y,\theta) = 1/\pi \sum_{k=-\infty}^{\infty} P^*(y-k)e^{-2j\,\theta k} P(y+k)$$

Skew estimation and correction are performed in the first stage of our system by using the projection profile technique and the WVD. The maximum intensity of the WVD of the horizontal histogram of a document image is used as the criterion for estimating its skew angle. The horizontal histogram of a properly oriented document page presents the maximum peaks and the most intent alterations between peaks and dips than any other histogram by angle of the same page. The WVD of the histograms represent their time-frequency distribution. The algorithm is able to detect skew angles varying between -89o to +89 in respect to the notional horizontal axis of properly oriented document.

The WVD and its maximum intensity are calculated for each histogram. Thus a curve of maximum intensity corresponds to each angle. The curve that presents the most maxima throughout the space domain is selected.

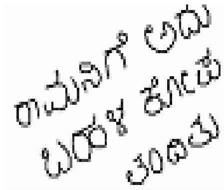


Figure 3.1 Image before Skew correction

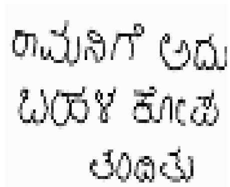


Figure 3.2 Images after Skew Correction

4. Slant Correction

The character segmentation, a pre-requisite stage in some systems, is a very difficult task of handwriting recognition. However, the slanted characters complicate it even more. A handwriting recognition system that does not deal with the slanted characters during the pre-processing stage requires much more data for it's training, in order to cover every case of inclined characters. In our system, we cope with this problem by automatically detecting the slant for each character and removing it. The same method as before is also used here, though the vertical projection profile of the word image is now used.

slant_angle=argmaxSD(vertical_projection_profile (horizontal_shear(I,θ)))

$$\theta min \leq \theta \leq \theta max$$

Where SD is the sum of the squares of the successive differences of the projection profiles, and search range is [-450, 450].

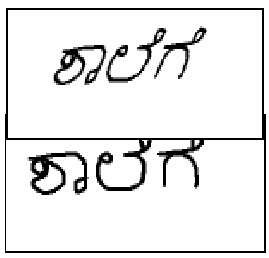


Figure 4.1 Slanted text, 4.2 Corrected slanted text

5. Character Segmentation

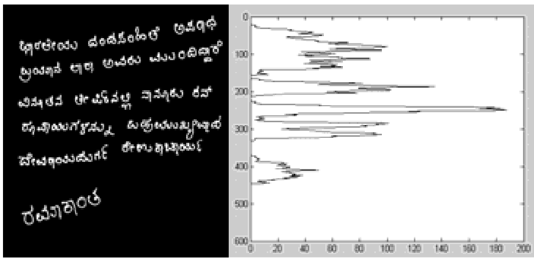


Figure 5.1 and 5.2 Input Document Image and corresponding horizontal histogram

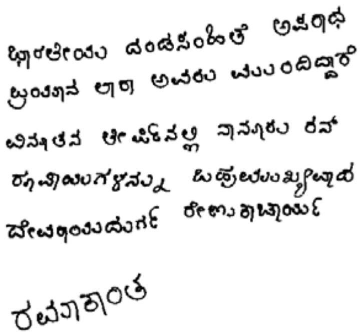


Figure 5.2 Line segmentation

5.1 Line Extraction

- Take the pre-processed image as the input.
- Draw the histogram of the number of pixels in each row.
- Find a row with no active pixel.
- If that row is followed by a number of zero-row pixels, where the number is large enough to classify them as intra- line spacing cut the image at that row. Let this point be the new row one of the document.
- For each such segmented line, call the word extraction module.
- When there are no more lines left, return to the calling function.

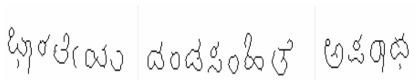


Figure 5.3 Extracted lines

5.2 Word Extraction

- Take the segmented line image as the input.
- Draw the histogram of the number of pixels in each column.
- Find a column with no active pixel.
- If that column is followed by a number of zero column pixels, where the number is large enough to classify them as intra-word spacing cut the image at that column. Let this point be the new column one of the line.
- Else, the spacing might be just an intra-character spacing. Ignore it. For each such segmented word, call the segment extraction module.
- When there are no more words left in the line, return to the calling function line extraction to get the next line.

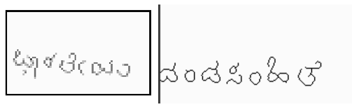


Figure 5.4 Extracted words

5.3 Character extraction

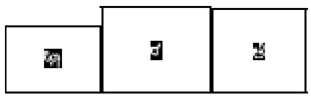


Figure 5.5 Extracted characters

6. Feature Extraction

Zernike moments are calculated for each of the image segment. The amplitudes of the Zernike moments are used as the feature vector.

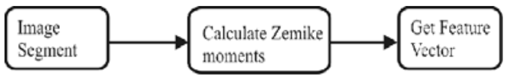


Figure 6.1 Feature extraction

6.1 Zernike Moments

Zernike Moments have been extensively used as the invariant global features for image recognition. Zernike moments have been analyzed and implemented by Teh and Chin [8], Bailey and Srinath [9] and Khotanzad and Hong [10]. These polynomials are a complete orthogonal basis set defined on unit disk x2+y2<=1. They are expressed as

$$A_{nm} = \frac{(n+1)}{\pi} \int\int F(x+y)[V_{nm}(x+y)]^{*} dxdy$$

$$\text{Where } x^2 + y^2 \le 1 \qquad (1)$$

n Any positive integer

m Any positive and negative integer subject to the constraints n-|m| = even and |m| = n

F(x,y) Function expressing binary/grey level image and * is complex conjugate

Zernike polynomials are expressed in polar co-ordinates as

$$Vnm(x, y)=Vnm(r, \theta)=Rnm(r).\exp(jrn\,\theta) \quad \text{-----------(2)}$$

r Length of vector from origin to (x,y)pixel

θ Angle between vector r and x-axis in counter clockwise

R_{nm} (r) *orthogonal* radial polynomial

Orthogonal radial polynomial is defined as

$$R_{nm}(r)= \sum_{s=0}^{(n-|m|)/2} (-1)^s F(n,m,s,r) \quad \text{------------------(3)}$$

$$F(n,m,s,r)= \frac{(n-s)!}{s!((n+|m|)/2-s)!(n-|m|)/2-s)!} \quad \text{------------------(4)}$$

For discrete image I(x,y) , the equation (1) becomes

$$A_{nm} = \frac{(n+1)}{\pi} \sum_x \sum_y I(x,y)\ [V_{nm}(x,y)]^* \quad \text{-----------------(5)}$$

$$\text{Where } x^2+y^2 \le 1$$

In order to compute Zernike moments of a discrete image, say, I(x,y), the image is first converted from Cartesian into polar co-ordinate system and mapped onto a unit disk. The center of the image is the center of unit disk. The pixels inside the unit disk are only used to compute Zernike moments and pixels outside the unit disk are not used for this purpose. Conversion from Cartesian to polar coordinate is given as

$$x=r\cos\theta \qquad y=r\sin\theta$$

$$r=\sqrt{(x^2+y^2)} \qquad \theta=\tan^{-1}(y/x) \quad \text{------------------(6)}$$

To achieve translation and scale invariance, the image is normalized using 0th and 1st order moments, which is given as

1).Translation invariance is achieved by moving the origin to center of image.

$$x' = M_{10}/M_{00} \text{ and}$$

$$y' = M_{01}/M_{00}$$

2). Scale invariance is achieved by altering each object so that its area (foreground pixels) is constant

$$M_{00}=\beta \quad \text{-------(7)}$$

Where β is constant and has predefined value. From equations (6) and (7), If G(x,y) represents new translated and scaled image

$$G(x,y)=F((x/a)+x',\ (y/a)+y') \quad \text{----------------------(8)}$$

$$\text{Where } a=\sqrt{(\beta/M_{00})}$$

6.2 Image Reconstruction Using Zernike Moments

It is possible to compute a discrete function F`(x,y), whose Zernike moments are same as that of F(x,y) up to order nmax. As n max approach infinity, F`(x,y) will become F(x,y)

$$F'(x,y)= \sum_{n=0}^{n_{max}} \sum_m A_{nm} V_{nm}(r,\theta) \quad \text{------------------------ (9)}$$

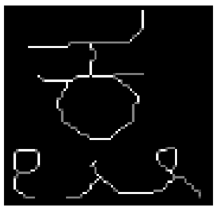
The values of n and me are same as in (1) and by solving (9), we have (10) as it is given in [15]

$$F'(x,y)= \sum_n \sum_{m>0} (C_{nm}\cos m\theta+S_{nm}\sin m\,\theta)R_{mn}(r)+C_{n0}R_{n0}(r)/2$$

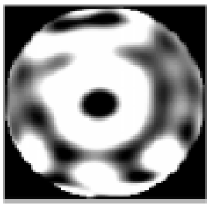
$$C_{nm}=2.\mathrm{Re}(A_{nm}) = \frac{2(n+1)}{\pi} \iint_{x^2+y^2\le 1} F(x,y)R_{nm}(r)\cos m\theta\,dxdy$$

$$S_{nm}=2.\mathrm{Im}(A_{nm}) = \frac{-2(n+1)}{\pi} \iint_{x^2+y^2\le 1} F(x,y)R_{nm}(r)\sin m\theta\,dxdy \quad \text{---(10)}$$

Shows the reconstruction of 64×64 handwritten Kannada image from Zernike moments.



a)Original Image



b) Image constructed from 2 to 13order Zernike moments

6.3 Zernike Moments as Feature Vector

Zernike moments may be used as feature vector for recognizing images. The magnitudes of these moments are rotational invariant and may be made translation and scale invariant using equ. no. (6) and (7) given in section 4. Due to this reason, Zernike moments may be used as a feature vector to recognize images particularly the hand printed ones. We have used these moments to recognize handwritten Kannada characters. The numbers of moments from order 0 to order 12 are 49. Zeroth order moment |A00| (number of foreground pixels of image) has been set to a constant value β (equ.no. 7) and is same for all images and the first order moment |A11| is zero for all the images. These two moments have not been used for image reconstruction and the inclusion of these two moments does not affect the image reconstruction. The total no. of Zernike moments and their corresponding possible feature vector for some maximum order are given in table 1.

Table1. Total No. Zernike Moments with possible feature Vector

Max. Order	Total No. of Zernike Moments	Possible Feature Vector
12	49	47
13	56	54
14	64	62
15	72	70
18	100	98

Only the use of lower order moments has been reported in literature for recognition purpose. The computation of higher order moments is very time consuming and it increases the time complexity of recognition algorithm. Therefore, the Zernike moments from order 2 to 13 are 56 and these may be used as feature vector.

7. Pattern Classification and Recognition

7.1 Support Vector Machines

Support vector machines (SVM) are a set of related supervised learning methods used for classification and regression. Their common factor is the use of a technique known as the "kernel trick" to apply linear classification techniques to non-linear classification problems. Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships.

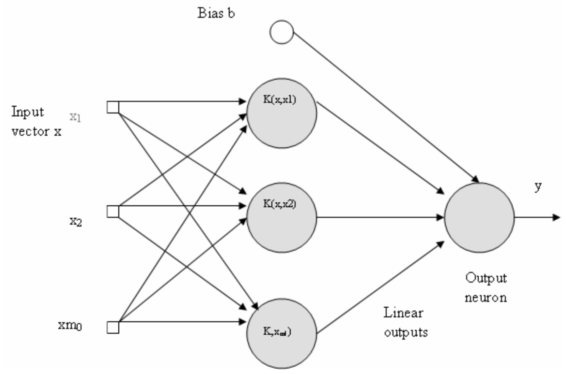


Figure 7.1 SVM architecture

7.2 Experimental Results for SVM using Zernike Features

A classification [11,12] task usually involves with training and testing data which consist of some data instances. Each instance in the training set contains one “target value” (class labels) and several “attributes” (features). The goal of SVM is to produce a model which predicts target value of data instances in the testing set which are given only the attributes. Given a training set of instance-label pairs (xi, yi), i = 1, . . . , where xi ∈ Rn and y ∈ 2 {1,−1} l, the support vector machines(SVM) [13] require the solution of the following optimization problem:

$$\min_{w,b,\xi} \quad \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i$$

$$\text{subject to} \quad y_i(w^T\phi(x_i)+b) \geq 1 - \xi_i,$$
$$\xi_i \geq 0$$

Here training vectors xi is mapped into a higher (maybe infinite) dimensional space by the function φ. Then SVM finds a linear separating hyper plane with the maximal margin in this higher dimensional space. C > 0 is the penalty parameter of the error term.

Furthermore, K(xi, xj) ≡ $\varphi(xi)^T$ φ(xj) is called the kernel function. The following are four basic kernels:

$$K(x_i, x_j) = x_i^T x_j$$

Polynomial

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$$

Radial Basis Function (RBF)

$$K(x_i, x_j) = \exp(-\gamma \, ||x_i - x_j||^2), \gamma > 0$$

Sigmoid :

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$$

Linear: Here γ, r *and* d are kernel parameters

7.3 Data Preprocessing

7.3.1 Scaling

The training data is scaled to be in the range of [0, 1] in order to avoid numerical problems. The test data is also scaled according to the parameters obtained during the training stage.

7.3.2 Model Selection

Though there are only four common kernels mentioned in Section 1, we must decide which one to try first. Then the penalty parameter C and γ kernel parameters are chosen.

7.3.3 Cross-validation and Grid-search [14]

The SVM algorithm with the Gaussian kernel requires two parameters: the parameter C > 0 is a regularization constant determining a trade-off between the empirical error (number of wrongly classified inputs) and the Gaussian kernel affects the complexity of the decision boundary. To select the best C and γ parameters for the training step, we used a cross validation via grid search. With this method, the training set is first separated into m folds. Sequentially a fold is considered as the validation set and the rest are for training. So an adaptive grid-search is performed on C and γ, trying exponentially growing sequences for the two parameters.

7.3.4 Results:

Classification of data using SVM with Radial features Without Cross validation
Nu = 1.000
Obj = -7.911703 rho=-0.03519
nSv= 8 nBSV= 8
Total nSv = 44
Accuracy = 72.7273% (8 / 11) classification
Mean Squared Error (MSE) = 3.72727
Squared Correction Coefficient (SCC) = 0.688783 (regression)
With Cross Validation
$ easy.py train.2.scale test.2.scale
Best C= 32.0 γ = 0.0078125 r = 40.9091

After applying the cross validation via grid search (see Figure 3) on a small randomly sampled subset of the training set (5%of the entire set), we have obtained the best rates for C = 32.0 and γ = 0.0078125 Finally, we trained the SVM classifier with the aforementioned parameters obtaining the following results on the test set:
Accuracy = 90.9091 %
Classification = 10/11
Mean Squared Error (MSE) =0.090909
Squared Correction Coefficient (SCC) = 0.993243

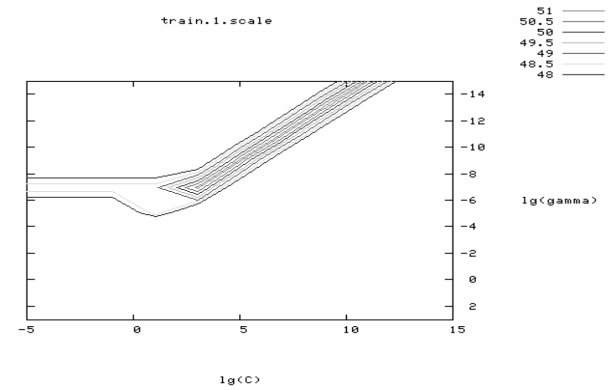


Figure 7.2 Graph obtained during the process of cross validation for Zernike features.

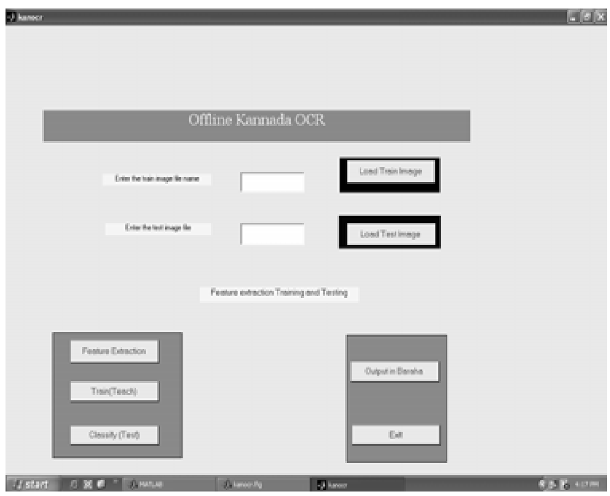


Figure 7.3 Initial GUI screenshot

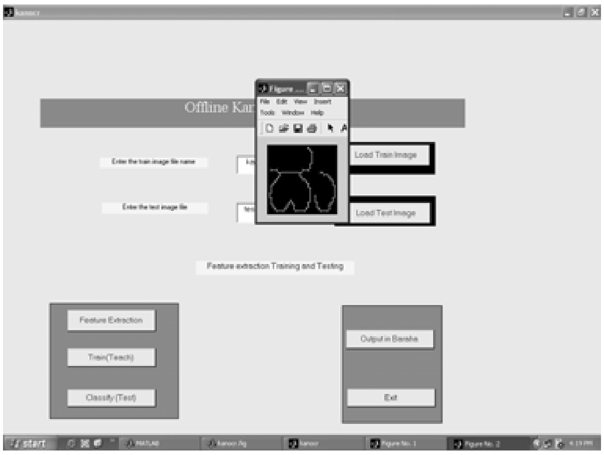


Figure 7.4 Screenshot during feature extraction

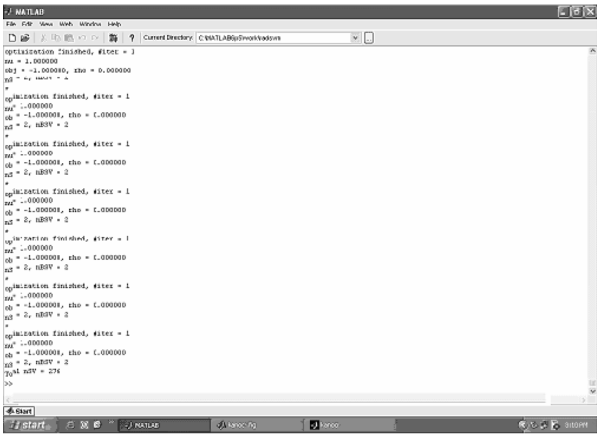


Figure 7.5 Screenshot during the process of Training

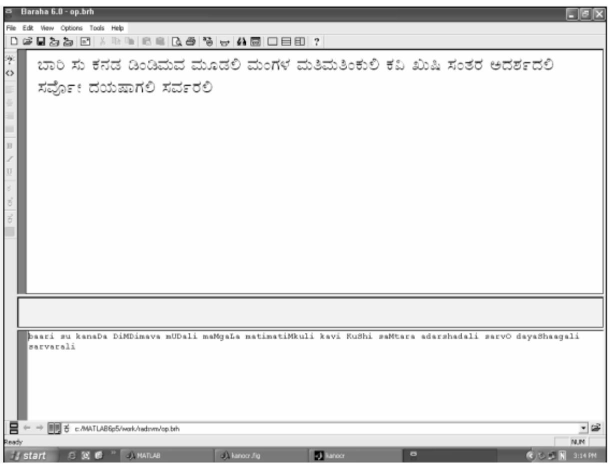


Figure 7.6 Screenshot of the output in Baraha

The feature vector constitutes the 56 features, due to Zernike moments from order 2 to 13 moments, used as input to train the classifier. The network was tested for handwritten Kannada character samples (not used for training) collected from different persons. The recognition rate achieved is approximately 94 %.

Conclusion

In this paper, we have given the details of Zernike moments and used them for the recognizing handwritten Kannada scripts. The recognition of handwritten Kannada characters is difficult task as compared to the machine printed characters due to large variation in writing styles of different persons. Due to rotational invariance of the magnitude of Zernike moments, they may be used to recognize handwritten characters. Variations in writing styles are covered by training the network with different handwritten samples collected from different persons. The rate of recognition can be further improved by training the network, adding more features and/or more Kannada handwriting samples.

References

[1] R. Plamondon and S. N. Srihari, "On-Line and Off- Line Handwriting Recognition: A Comprehensive Survey", IEEE trans. Pattern Analysis and Machine Intelligence, 22 (1), 63-84, 2000.

[2] A.L. Koerich, R. Sabourin and C.Y. Suen, "Large vocabulary off-line handwriting recognition:A survey", Pattern Analysis Applications 6(2003), 97-121.

[3] T.V.Ashwin, P.S.Sastry, "A font and size-independent OCR system for printed Kannada documents using support vector machines", sadhana, Vol.27, part1, Feb. 2002, pp., 35-58

[4] E Kavallieratou, N Fakotakis and G Kokkinakis , " An integrated system for handwritten Document image processing", Int. Journal of pattern Recognition and Artificial Intelligence, vol.17, no. 4(2003) 617-636

[5] E Kavallieratou, N Fakotakis and G Kokkinakis, " Skew angle estimation for printed and handwritten documents using the Wigner-Ville distribution", Image and Vision Computing 20 813-824, Elsevier Science B

[6] B.Yegnanarayana, Artficial Neural Network, PHL, 1999

[7] C. Singh and S. Kumar, "Multilayer Feedforward Back-Propagation Neural Network Based Classifier forHandwritten Pattern Classification - An Analysis for Handwritten Devanagari Numerals", Presented in 91st Indian Science Congress, January 3-7,2004, Panjab University, Chandigarh, India.

[8] C.H. Teh and R.T. Chin, "On Image Analysis by Method of Moments", IEEE Trans On Pattern Analysis and Machine intelligence Vol.10, No. 11, July 1988.

[9] R.R. Bailey and M. Srinath, "Orthogonal Moment Feature for Use with Parametric and Non-Parametric Classifiers", IEEE Trans. for Pattern Analysis and Machine Intelligence, vol. 18, pp. 389-396,1996.

[10] A. Khotanzad and Y.H. Hong, "Invariant Image Recognition by Zernike Moments", IEEE Trans On PatternAnalysis and Machine Intelligence Vol. 12 , No. 5, May 1990.
[11] Heloise Hse, A. Richard Newton. Sketched Symbol Recognition using Zernike Moments
[12] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, A Practical Guide to Support Vector Classification
[13] Chang, C.-C. and C.-J. Lin (2001). LIBSVM: a library for support vector machines
[14] A Practical Guide to Support Vector Classification Chih Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin Department of Computer Science and Information Engineering National Taiwan Uni versity Taipei 106, Taiwan - cjlin@csie.ntu.edu.tw.

Thungamani.M received B.E from Visvesvaraya Technological University and M.Tech in Computer Science and Engineering in the year 2007.She has experience of around 08 years in academics. Currently pursuing Ph.D. and working as Research Assistant in Tumkur University, Tumkur. Life membership in Indian Society for Technical Education (MISTE) The Institution of Electronics and Telecommunication Engineers (IETE).

Dr. Ramakanth Kumar P completed his Ph.D. from Mangalore University in the area of Pattern Recognition. He has experience of around 16 years in Academics and Industry. His areas of interest are Image Processing, Pattern Recognition and Natural Language Processing. He has to his credits 03 National Journals, 15 International Journals, 20 Conferences. He is a member of the Computer Society of India (CSI) and a life member of Indian Society for Technical Education (ISTE). He has completed number of research and consultancy projects for DRDO.

Keshava Prasanna received B.E from Bangalore University and M.Tech in Information and Technology in the year 2005.He has experience of around 13 years in academics. Currently pursuing Ph.D. and working as Research Assistant in Tumkur University, Tumkur. Life membership in Indian Society for Technical Education (ISTE).

Shravani Krishna Rau pursuing B.E degree in Computer Science & Engineering from Visvesvaraya Technological University. Student of R.V.College of Engineering, Bangalore.