

A simple and efficient optical character recognition system for basic symbols in printed Kannada text

R SANJEEV KUNTE and R D SUDHAKER SAMUEL

Department of Electronics and Communication, S J College of Engineering,
Mysore 570 006
e-mail: sanjeevkunte@yahoo.com

MS received 18 May 2006; revised 6 August 2006

Abstract. Optical Character Recognition (OCR) systems have been effectively developed for the recognition of printed characters of non-Indian languages. Efforts are on the way for the development of efficient OCR systems for Indian languages, especially for Kannada, a popular South Indian language. We present in this paper an OCR system developed for the recognition of basic characters (vowels and consonants) in printed Kannada text, which can handle different font sizes and font types. Hu's invariant moments and Zernike moments that have been progressively used in pattern recognition are used in our system to extract the features of printed Kannada characters. Neural classifiers have been effectively used for the classification of characters based on moment features. An encouraging recognition rate of 96.8% has been obtained. The system methodology can be extended for the recognition of other south Indian languages, especially for Telugu.

Keywords. Kannada text; OCR; invariant moments; neural networks.

1. Introduction

Optical Character Recognition (OCR) is one of the oldest sub fields of pattern recognition with a rich contribution for the recognition of printed documents. OCR systems scan the documents printed on a paper as an image and recognize the characters present in the document image to form a separate digital text document, which can be edited or processed.

Due to the impact and the advancements in the Information Technology, nowadays more emphasis is given in Karnataka to use Kannada at all levels and hence the use of Kannada in computer systems is also a necessity. Therefore, efficient OCR systems for Kannada are one of the present day requirements. Currently there are many OCR systems available for handling printed 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 especially for a South Indian language like Kannada.

In the Indian OCR context, most of the works have been carried out for the OCR for Devanagari, Bangla and Telugu Scripts and not many works are reported for Kannada language.

Moment based features are a traditional and widely used tool for character recognition. Classical moment invariants were introduced by Hu (1962) which is invariant under translation, rotation and scaling. However, Hu's moments are not derived from family of orthogonal functions, and so contain much redundant information about a character's shape. Hence, Zernike moments based on the theory of orthogonal polynomials are becoming popular for character recognition nowadays.

RBF Neural networks have fast training/learning rate because of their local-tuned neurons (Moody & Darken 1989). They also exhibit universal approximation property and have good generalization ability (Park & Wsandberg 1991). Hence, they have been used in the field of character recognition systems.

The successful application of moment features and neural classifiers in character recognition systems has motivated this work. In this paper we have presented an OCR system for basic Kannada characters, selecting the scheme of feature extraction using the moments and RBF neural networks as classifiers to identify and classify the characters. Hu's invariant moments and Zernike moments are considered for feature extraction.

The novelty of the proposed recognition scheme lies in its simplicity because of the use of moment features which are simple and easy to calculate. Beside simplicity, the proposed method is highly efficient as the experimental results show an encouraging recognition rate of 96.8%.

The proposed system classifies only basic Kannada characters (vowels + consonants). However, the recognition scheme developed is capable of classifying complex and similar shaped Kannada characters, which is evident from the Scatter plot shown in figure 9. The complete OCR system is a simple extension of the proposed recognition scheme with the inclusion of other characters along with a suitable classifier.

The organization of the paper is as follows: Literature survey is presented in § 2. In § 3 a brief description about Kannada language and its character set is presented. The basics of moments are presented in § 4 and 5. The overall recognition scheme used in the system is presented in § 6. Segmentation of characters from the document image is discussed in § 7. The method of feature extraction is presented in § 8. The architecture of RBF neural networks used as classifiers is presented in § 9. Experimental results and conclusions are discussed in § 10 and 11.

2. Literature survey

Some of the schemes that are reported in the recent works for character recognition in Kannada and another south Indian language Telugu (having character set similar to Kannada language) are as follows:

Nagabhushan & Pai (1999) have proposed a modified region decomposition method and optimal depth decision tree for the recognition of printed Kannada characters. In their work, the basic and consonant-vowel combination characters are only considered. The conjunct combination characters, which are complex and large in numbers, are not considered for recognition.

In a Kannada OCR system reported by Ashwin & Sastry (2002) the character image is split into three basic zones. Each zone is divided into a number of circular tracks and sectors and feature like the number of ON pixels in each annular region is used for classification using Support Vector Machine.

Negi *et al* (2001) have proposed an OCR system for Telugu characters based on Fringe distance measurement. Template of all the characters to be recognized by the system is

ಅ ಆ ಇ ಈ ಉ ಊ ಋ ೠ ಎ ಏ ಐ ಒ ಓ ಔ ಋ :
 a ā i ī u ū ṛ ṝ e ē ai o ō au am ah

Figure 1. Kannada vowels.

formed. A type of similarity measure is performed between the test character and the templates. The template yielding the highest similarity score is declared as the class of the test character.

In a Bilingual OCR for Hindi–Telugu documents reported by Jawahar *et al* (2003) the entire image of the character is used as a feature vector followed by dimension reduction using Principal component analysis and later classified by Support Vector Machine.

3. Kannada script

Kannada is one of the four popular Dravidian languages of South India. Kannada is written horizontally from left to right and the concept of lower and upper case is absent. Kannada is a non-cursive script. That is, a Kannada word is written without joining the characters of the word. The characters are isolated within a word.

Kannada language has 16 vowels and 34 consonants as the basic alphabet of the language as shown in figures 1 and 2 respectively.

Each vowel has a vowel sign (modifier) and each consonant has a basic form (primitive). A basic form of consonant can combine with the vowel sign to form another set of 16 Consonant–Vowel (CV) composite characters called as ‘gunithakshara’. In Kannada, all the 34 consonants have a half/short form, called as ‘Vatthus’, which can be referred as half consonants or subscripts. Any half consonant can appear below any other consonant or a CV character as subscript character to form a conjunct-consonant character.

ಕ	k	ಖ	kh	ಗ	g	ಘ	gh	ಜ	ñ
ಚ	c	ಛ	ch	ಜ	j	ಝ	jh	ಞ	ñ
ಟ	t	ಠ	th	ಡ	d	ಢ	dh	ಣ	ṇ
ತ	t	ಥ	th	ದ	d	ಧ	dh	ನ	n
ಪ	p	ಫ	ph	ಬ	b	ಭ	bh	ಮ	m
ಯ	y	ರ	r	ಲ	l	ವ	v	ಶ	ś
ಷ	ṣ	ಸ	s	ಹ	h	ಳ	l		

Figure 2. Kannada consonants.

4. Hu's invariant moments

The general form of a regular moment function m_{pq} of order $(p + q)$ of an image intensity function $f(x, y)$ can be given as,

$$m_{pq} = \iint x^p y^q f(x, y) dx dy \quad (1)$$

For a digital image, the central moments, which are invariant to translation, are defined as:

$$\mu_{pq} = \sum_x \sum_y (x - x_c)^p (y - y_c)^q f(x, y) \quad (2)$$

where, $x_c = m_{10}/m_{00}$ and $y_c = m_{01}/m_{00}$ are the co-ordinates of the centroid.

Hu (1962) introduced 7 non-linear functions defined on regular moments which are invariant to rotation, scaling and translation. They are given by,

$$\Phi_1 = \nu_{20} + \nu_{02}$$

$$\Phi_2 = (\nu_{20} - \nu_{02})^2 + 4\nu_{11}^2$$

$$\Phi_3 = (\nu_{30} - 3\nu_{12})^2 + (3\nu_{21} - \nu_{03})^2$$

$$\Phi_4 = (\nu_{30} + \nu_{12})^2 + (\nu_{21} + \nu_{03})^2$$

$$\Phi_5 = (\nu_{30} - 3\nu_{12})(\nu_{30} + \nu_{12})[(\nu_{30} + \nu_{12})^2 - 3(\nu_{21} + \nu_{03})^2] \\ + (3\nu_{21} - \nu_{03})(\nu_{21} + \nu_{03})[3(\nu_{30} + \nu_{12})^2 - (\nu_{21} + \nu_{03})^2]$$

$$\Phi_6 = (\nu_{20} - \nu_{02})[(\nu_{30} + \nu_{12})^2 - (\nu_{21} + \nu_{03})^2] + 4\nu_{11}(\nu_{30} + \nu_{12})(\nu_{21} + \nu_{03})$$

$$\Phi_7 = 3(\nu_{21} - \nu_{03})(\nu_{30} + \nu_{12})[(\nu_{30} + \nu_{12})^2 - 3(\nu_{21} + \nu_{03})^2] \\ - (\nu_{30} - 3\nu_{12})(\nu_{21} + \nu_{03})[3(\nu_{30} + \nu_{12})^2 - (\nu_{21} + \nu_{03})^2] \quad (3)$$

where, $\nu_{pq} = \frac{\mu_{pq}}{\mu_{00}^{(1+(p+q)/2)}}$, $p + q \geq 2$.

The definition of regular moments has the form of projection of $f(x, y)$ function onto the basis set $x^p y^q$. Since, the basis set $x^p y^q$ is not orthogonal (Teague 1980), the recovery of image from these moments is difficult and has a drawback of information redundancy.

5. Zernike moments

Zernike moments are due to Zernike polynomials introduced by Zernike (1934). Zernike polynomials are a set of complex polynomials which form a complete orthogonal set over interior of the unit circle. The Zernike moments are projections of the input image onto the space spanned by the orthogonal V functions:

$$V_{nm}(x, y) = V_{nm}(r, \theta) = R_{nm}(r) \cdot \exp(jm\theta) \quad (4)$$

where, $n \geq 0$, $|m| \leq n$, $n - |m|$ is even, r is the length of vector from origin to (x, y) pixel and θ is angle between vector r and the x -axis in counter-clock direction.

The orthogonal radial polynomial $R_{nm}(r)$ is defined as:

$$R_{nm}(r) = \sum_{s=0}^{(n-|m|)/2} (-1)^s g \frac{(n-s)!}{s! \left(\frac{n+|m|}{2} - s\right)! \left(\frac{n-|m|}{2} - s\right)!} g^{r(n-2s)} \quad (5)$$

The Zernike moment of order of n with a repetition m of a continuous image function $f(x, y)$ is given by:

$$Z_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) [V_{nm}(x, y)]^* \quad (6)$$

where, $x^2 + y^2 \leq 1$ and the symbol $*$ denotes the complex conjugate operator.

In order to compute the Zernike moments of a given image, the center of the image is taken as the origin and image coordinates are mapped to the range of the unit circle $x^2 + y^2 \leq 1$. Only those pixels falling inside the unit circle are considered for the moment computation.

If the image is rotated by an angle α , the transformed Zernike moment function Z'_{nm} are given by:

$$Z'_{nm} = Z_{nm} \cdot e^{-jm\alpha} \quad (7)$$

This means that the magnitude of the moments stays the same after the rotation. Hence, the magnitude of the Zernike moments of the image, $|Z_{nm}|$, could be taken as rotation invariant features (Khotanzad 1998).

Zernike moments have the following advantages:

- The magnitudes of Zernike moments are invariant to rotation
- Zernike moments are robust to noise and minor variations in shape (Khotanzad 1998)
- Since the basis is orthogonal, Zernike moments have minimum information redundancy (Teague 1980)

However, the computation of Zernike moments poses several problems (Mukundan *et al* 2001) such as:

- The coordinates of the image must be mapped to unit circle
- Computational complexity of radial Zernike polynomial increases as the order becomes large.

6. Recognition scheme

The block schematic of the recognition system developed is as shown in the figure 3, which is in line with the normally used approach for OCR systems. The input to the system is a digital

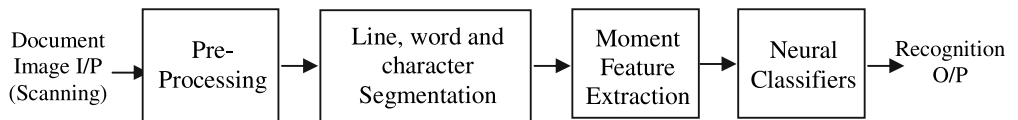


Figure 3. Kannada OCR system block schematic.

image of the document containing printed Kannada text captured by scanning the document using a flatbed scanner. It is subjected to pre-processing to eliminate noise in the document and segmentation, to separate out the characters in the image. A segmented character may be a vowel, a consonant, a consonant–vowel composite or a Vatthu. From the segmented character set, only vowels and consonant characters are separated manually and stored in test data file for recognition.

For a given test character, moment features (both Hu's and Zernike moments) are extracted. The pre-trained neural classifier recognizes the character using the moment features and generates the class information. The character class is mapped to the Kannada font and it is stored in a document file. The output of the system is an electronic version of the input document (with only vowels and consonants). Also, the recognized character is displayed on the system display screen as a sort of feedback to the user.

7. Segmentation

After scanning the document, the document image is subjected to pre-processing for background noise elimination, skew correction and binarization to generate the bit map image of the text. The pre-processed image is then segmented into lines, words and characters. The details of line, word and character segmentation are discussed in the following sub-sections.

7.1 Line segmentation

To separate the text lines, from the document image, the horizontal projection profile of the document image is found. The horizontal projection profile is the histogram of the number of ON pixels along every row of the image. White space between text lines is used to segment the text lines. Figure 4 shows a sample Kannada document along with its horizontal projection. The projection profile will have valleys of zero height between the text lines. Line segmentation is done at these points.

7.2 Word segmentation

The spacing between the words is used for word segmentation. For Kannada script, spacing between the words is greater than the spacing between characters in a word. The spacing between the words is found by taking the vertical projection profile of an input text line. Vertical projection profile is the sum of ON pixels along every column of the image.

A sample input text line and its vertical projection profile is shown in figure 5. From the vertical projection profile it can be observed that, the width of zero-valued valleys is more between the words in the line as compared to the width of zero-valued valleys that exists

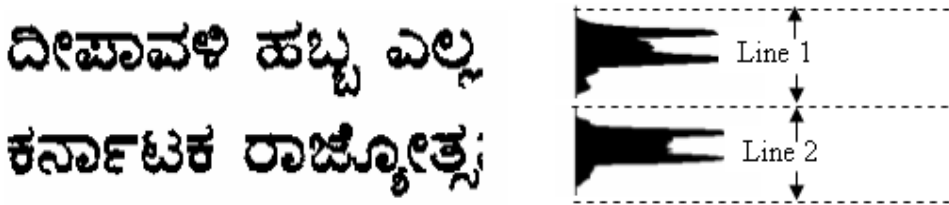


Figure 4. Text lines with horizontal projection profiles and line segmentation details for a sample Kannada text document.

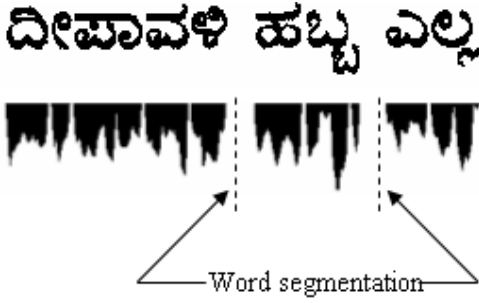


Figure 5. A text line and its vertical projection profile indicating word segmentation for a sample document.

between characters in a word. This information is used to separate words from the input text lines.

7.3 Character segmentation

As Kannada is a non-cursive script, the individual characters in a word are isolated. Spacing between the characters can be used for segmentation. But, sometimes in the vertical projection profile of a word, there will not be any zero-valued valleys, due to the presence of conjunct-consonant (subscript/Vatthu) characters. The subscript character position overlaps with the two adjacent main characters in vertical direction as shown in figure 6.

In these cases the usual method of vertical projection profile to separate characters is not possible.

Hence, for character segmentation, it is first checked for the presence of subscripts in a word (Kunte & Samuel 2006). If subscripts are present, they are extracted first from the word using connected component method (Gonzalez & Woods 1993). The remaining characters from the word are extracted using vertical projection profile.

8. Feature extraction

In the proposed method, two types of features based on (i) Hu's invariant moments and (ii) Zernike moments are extracted from segmented characters.

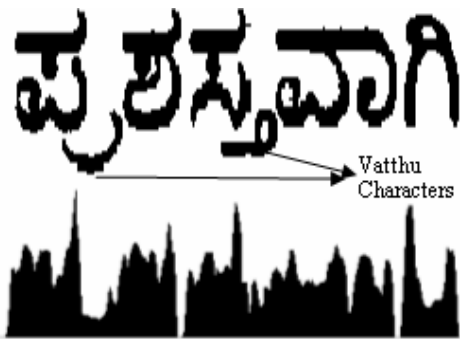


Figure 6. A sample Kannada word with Vatthus along with its vertical projection profile.

8.1 Hu's invariant moments

From the pre-processed segmented character, seven moments Φ_1 to Φ_7 and are calculated using (3). The feature vector F_1 as given in (8) forms Hu's moment feature.

$$F_1 = [\Phi_1, \Phi_2, \Phi_3, \Phi_4, \Phi_5, \Phi_6, \Phi_7] \quad (8)$$

8.2 Zernike moments

Binary image of pre-processed segmented character is mapped to the unit circle. For desired value of order of moment i.e., n , real and imaginary parts of the Zernike moments are computed from (6) using the radial polynomial from (5). A feature vector F_2 consisting of computed moments forms Zernike moment feature.

9. RBF neural classifiers

Radial basis function (RBF) neural networks have recently attracted extensive research interests because, they are universal approximators (Park & Wsandberg 1991), their learning speed is fast because of locally-tuned neurons (Moody & Darken 1989), they have more compact topology than other neural networks (Girosi & Poggio 1990). Radial basis function neural network is used for a wide range of application primarily because it can approximate any regular function and its training speed is faster than that of a multi-layer perceptron.

9.1 RBF network architecture

RBF network consists of three layers, namely; input layer, hidden layer and the output layer. Each node in the input layer corresponds to a component of the feature vector F . The second layer, the only hidden layer in the network applies non-linear transformation from input space into hidden space by employing non-linear activation function, such as Gaussian kernel. Linear nodes at the output layer correspond to the possible classes of the problem.

In the proposed work, two RBF networks were created corresponding to Hu's invariant moments and Zernike moments.

RBF network for Hu's invariant moments (*RBF-Hu*) consists of 7 nodes in the input layer corresponding to 7 moment features computed from the image of segmented Kannada character. In the RBF network of Zernike moments (*RBF-Zernike*), the number of nodes in the input layer were varied from 7 to 34 depending upon the order of Zernike moment chosen for feature extraction. However, in both the RBF networks, the number of nodes in the output layer was 49 corresponding to 49 classes of basic Kannada characters. The number of radial basis function (Gaussian) in the hidden layer in both RBF networks was chosen to be equal to number of training samples (2500).

10. Experimental results and discussion

The data set for training the RBF network is formed from, about 50 different samples of each basic Kannada character (with different fonts and font size varying from 12 to 18 points) to a total of about 2500 characters.

A test set was created with about 20 different samples (other than training samples) of each basic Kannada character to a total of about 1000 characters. Another set of about 1000 characters from different Kannada newspapers and magazines were included in the test set.

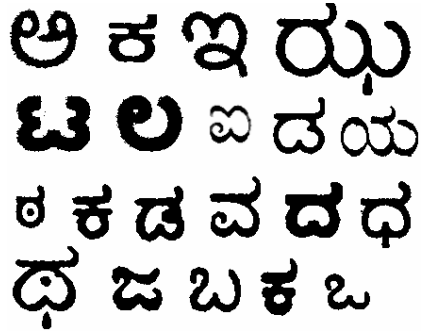


Figure 7. Sample test data.

Figure 7 shows sample test data containing mixture of characters of different font type and sizes (segmented from different documents) used for recognition.

Due to close resemblance in shapes, some of the characters were misclassified during recognition. Table 1 lists the confused character pairs, where the top row corresponds to original characters and the bottom row indicates the corresponding confused characters.

10.1 Hu's moments

A set of seven Hu's invariant moments were extracted from the pre-processed test characters. These features were applied to pre-trained neural network *RBF-Hu*. A recognition rate of about 82% was obtained.

10.2 Zernike moments

From the pre-processed test characters Zernike moments of varying order are extracted. The moments of order 4 to 10 are used for the experiment. The corresponding numbers of moments computed are 7, 10, 14, 18, 23, 28 and 34. These moment features were applied to pre-trained neural network *RBF-Zernike*. The recognition rates obtained for different order of moments are listed in table 2.

Table 1. Confused character pairs.

ಅ	ರ	ಧ	ಡ	ದ	ಪ
ಆ	ಠ	ಢ	ಡ	ಧ	ಫ

Table 2. Recognition rate for various orders of Zernike moments.

Moment order	4	5	6	7	8	9	10
No. of moments	7	10	14	18	23	28	34
Recognition rate	91.8%	92.5%	94.2%	95.7%	96.3%	96.7%	96.8%

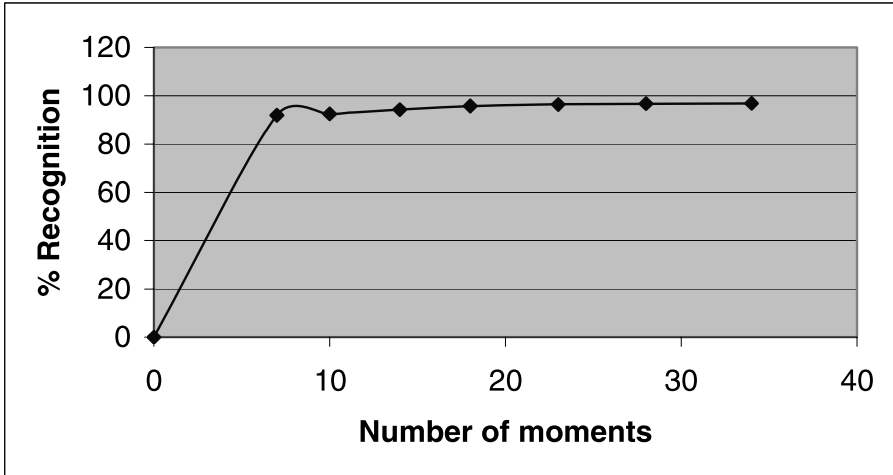


Figure 8. Recognition rate/number of Zernike moments.

A plot of recognition rate versus order of number of Zernike moments in figure 8 shows only a marginal improvement in recognition rate for the number of moment features above 20.

When compared with Hu's moments, the Zernike moments for the same number of features (i.e., number of moments 7), recognition rate is much higher. This is expected as the Hu's moments are not orthogonal and hence they are susceptible to linear transformations.

10.3 Clustering of resembling character classes in feature space

To study how the different character classes are clustered in the feature space, 3 samples of some of the resembling characters were selected randomly from the test data file and features were extracted using Zernike moments up to 10th order (to a total of 34 features). For each sample, the features were reduced from 34 to 2 using *Multi-dimension scaling*. A plot of these samples in feature space using the reduced two features is shown in figure 9.

It can be seen that in the feature space the samples of same class are clustered more tightly. At the same time the samples such as \varnothing and \emptyset which have close resemblance in shape, are clustered at a good distance from each other. This shows that the proposed feature extraction scheme using Zernike moments exhibits excellent between-class distance and smaller within-class scatter.

10.4 Comparative study

In this subsection a comparative study of the proposed method with three existing methods proposed by Ashwin & Sastry (2002), Nagabhushan & Pai (1999) and VijayaKumar & Ramakrishnan (2004) with respect to character segmentation, recognition rate, type of feature, font size and font type independence is presented.

In OCR system for Kannada by Ashwin & Sastry (2002), the following features are comparable with the system proposed in this paper.

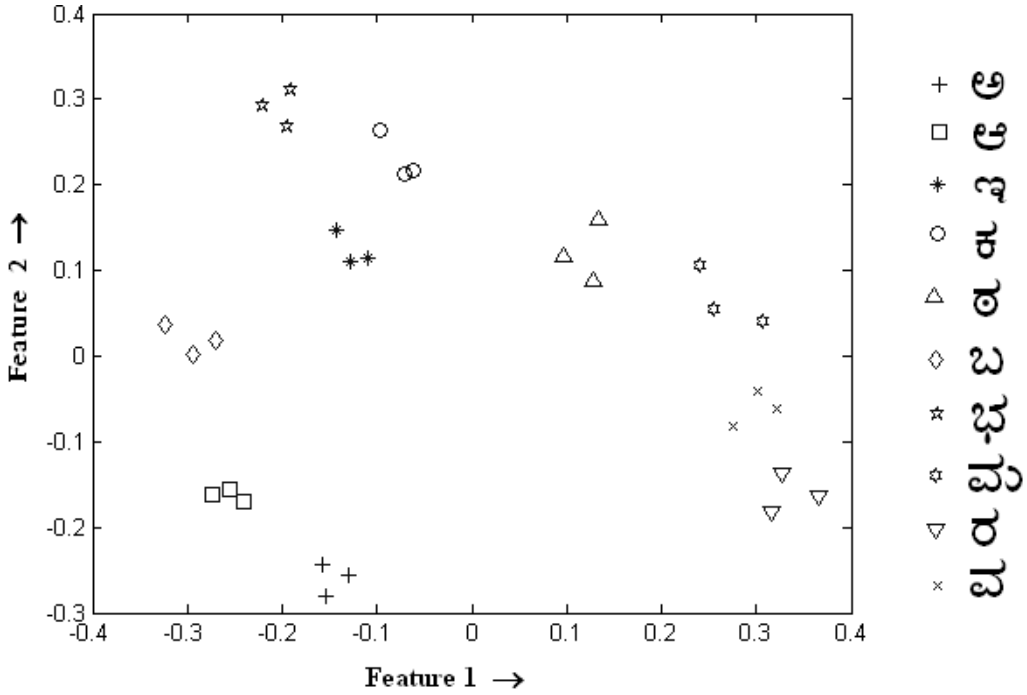


Figure 9. Distribution of character classes in feature space.

- (i) They have proposed to chose the individual recognition units as the constituents of a character (i.e., the base consonant, the vowel modifier and the consonant conjunct) but not as a whole character (*akshara*). This is due to the large number of character combinations possible, building a classifier to recognize a whole character is difficult.
- (ii) They propose to use the structural features because computing the Zernike moments takes more than eight times as much computational time as computing the structural features.

But, it is possible to build a two-stage classifier to recognize the large set of character combination. The consonant–vowel composite characters of a consonant are all grouped together as a single class of character in the first stage of classification. Later, in the second stage, a character belonging to such a group is further classified within its group by a separate classifier meant for classifying the consonant group of characters. Hence, it is possible to use a whole character itself as a recognition unit for classification.

The character segmentation method proposed by Ashwin & Sastry (2002) leads to segmentation errors, which reduces the recognition rate. The best recognition rate obtained in their case was 86.11%. However, in the proposed method, the individual recognition unit were taken as a whole character. Along with a combination of Zernike features the proposed method has obtained recognition rate of 96.8%.

Presently many algorithms are available for fast computation of Zernike moments (Chong *et al* 2003) which work almost 20 times faster (Mohammed & Yang 2002) than the normal algorithms. So, computation of Zernike moments are much more easier than computing modified structural features proposed by Ashwin & Sastry (2002).

This justifies the proposed method to be simple because of using features like Zernike moments which is highly efficient (recognition rate 96.8%) due to the character segmentation method used.

In another OCR system for Kannada reported by Nagabhushan & Pai (1999), the following features are comparable with the system proposed in this paper:

- (i) The data set used for training and testing the OCR system were same and a small set of 150 characters were considered for experimentation resulting in 100% recognition rate.
- (ii) The characters used for experimentation were assumed to be of particular font size and type.

However, in the proposed method a large data set was used and the testing data set was drawn from samples other than training set. Further, the proposed system is capable of handling different Kannada fonts and font sizes varying from 14 to 18 points.

In yet another OCR system for Kannada reported by VijayKumar & Ramakrishnan (2004), the following features are comparable with the system proposed in this paper:

- (i) The best possible recognition rate was obtained using 60 features.
- (ii) Classification time alone itself was 1.69 min for 1453 characters on a Sun Ultra Sparc workstation.

In the proposed method, best possible recognition rate was obtained with just 23 features. The computational time taken for feature extraction and classification of 2000 pre-segmented characters was 1.23 min. The system was implemented on a P IV, 2.1 GHz computer using VC++.

10.5 *Future work*

At present, the system is limited to handle basic Kannada characters only. However, the recognition scheme developed is capable of classifying complex and similar shaped Kannada characters. The system can be further developed to a complete OCR system by including all the characters from Kannada alphabet set and a proper classifier to handle large set of characters. The recognition rate of the proposed system can be improved for the recognition of confused characters by adding structural features.

11. **Conclusion**

In this paper, a simple and efficient OCR system for basic Kannada characters employing moment features is presented. The system methodology is capable of handling different font sets and sizes. Telugu language, which is another popular South Indian language, has character set very similar to Kannada. Hence, the methodology developed can be extended to Telugu OCR system and even for other South Indian languages. With further developments in the system design, by including a two-stage classifier, the system can handle the full set of Kannada characters (by including the Consonant–Vowel composite characters and subscript characters) to become a complete OCR system for Kannada language.

This work was supported in part by research grants from All India Council for Technical Education (AICTE), New Delhi, under Research promotion scheme.

References

- Ashwin T V, Sastry P S 2002 A font and size-independent OCR system for printed Kannada documents using support vector machines. *Sādhanā* 27: 35–58
- Chong Chee-Way, Raveendran P, Mukundan R 2003 A comparative analysis of algorithms for fast computation of Zernike moments. *Pattern Rec.* 36: 731–742
- Girosi F, Poggio T 1990 Networks and the best approximation property. *Bio. Cybernetics.* 63: 169–176
- Gonzalez R C, Woods R E 1993 *Digital image processing* (Boston, MA, USA: Addison Wesley Longman Publishing Co. Inc.)
- Hu M-K 1962 Visual pattern recognition by moment invariants. *IRE Trans. Inf. Theory.* IT-8: 179–187
- Jawahar C V, Pavan Kumar, Ravi Kiran S S 2003 A Bilingual OCR for Hindi-Telugu documents and its applications. *Proc. Seventh Int. Confer. on Document Anal. and Rec.* 408–412
- Khotanzad A 1998 Rotation invariant pattern recognition using Zernike moments. *Proc. Int. Conf. on Pattern Rec.* 326–328
- Kunte Sanjeev R, Sudhaker Samuel R D 2006 A two-stage character segmentation scheme for Printed Kannada text. *J. Graphics, Vision and Image Processing* 6: 1–8
- Moody J, Darken C J 1989 Fast learning in network of locally-tuned processing units. *J. Neural Comput.* 1: 281–294
- Mukundan R, Ong S H, Lee P A 2001 Image analysis by Tchebichef moments. *IEEE Trans. Image Processing* 10: 1357–1364
- Mohammed Al-Rawi, Yang Jie 2002 Practical fast computation of Zernike moments. *J. Comput. Sci. and Technol.* 17: 181–188
- Nagabhushan P, Pai Radhika M 1999 Modified region decomposition method and optimal depth decision tree in the recognition of non-uniform sized characters — An experimentation with Kannada characters. *Pattern Rec. Lett.* 20: 1467–1475
- Negi Atul, Chakravarthy Bhagavathi, Krishna B 2001 An OCR system for Telugu. *Proc. Sixth Inter. Confer. on Document Anal. and Rec.* 1110–1114
- Park J, Wsandberg J 1991 Universal approximation using radial basis function neural networks. *J. Neural Comput.* 1: 246–257
- Teague MR 1980 Image analysis via the general theory of moments. *J. Optical Soc. Amer.* 70: 920–930
- VijayaKumar B, Ramakrishnan A G 2004 Radial basis function and sub-space approach for printed Kannada text recognition. *Proc. IEEE ICASSP 2004* 5: 321–324
- Zernike F 1934 *Physica.* 1: 689–704