IOSR Journal of Engineering
Apr. 2012, Vol. 2(4) pp: 697-703

Automatic Recognition of isolated Marathi Handwritten Numerals

Ms. Pratibha.V. Waje[1], Ms.Mrunalinee N. Shirole[2], Prof.Sandeep Sahu[3], Dr.R.S.Patil[4]

[1](Software System, Shriram Institute of Technology, Jabalpur, India)
[2](CSE, D.Y. Patil College of Engg.Pune, India)
[3](A.P(CSE), Shriram Institute of Technology, Jabalpur, India)
[4](Professor, Sandip Institute of Technology & Research center, Nasik, India)

ABSTRACT

In this paper, we present a method for automatic recognition of Marathi handwritten numerals. The aim of handwritten numeral recognition (HNR) system is to classify input numeral as one of K classes. Handwritten numeral recognisation system have two component feature analysis & pattern classification The feature extraction is by using the chain code and Fourier Descriptor we determine the shape of numerals . After preprocessing the numeral image, the normalized chain code and the Fourier descriptors of the contour of the numeral are extracted. These features are then fed in the Support Vector Machine (SVM) for classification. The proposed method is experimented on a database of 12690 samples of Marathi handwritten numeral using fivefold cross validation technique. We have obtained recognition accuracy of 98.15%.

***Keywords* - Handwritten numerals recognition, Fourier descriptors, chain code, Support Vector Machines.**

I. INTRODUCTION

The aim of handwritten numeral recognition (HNR) system is to classify input numeral as one of K classes. Conventional HNR systems have two components: feature analysis and pattern classification, as shown in Fig. 1. In Feature analysis step, information relevant for pattern classifier. The pattern classification step labels the numeral as one of K classes using the class models. Over the years, considerable amount of work has been carried out in the area of HNR. Various methods have been proposed in the literature for classification of handwritten numerals. These include Hough transformations, histogram methods, principal component analysis, and support vector machines, nearest neighbor techniques, neural computing and fuzzy based approaches. An extensive survey of recognition performance for large handwritten database through many kinds of features and classifiers is reported. In comparison with HNR systems of various non Indian scripts [e.g. Roman, Arabic, and Chinese] we find that the recognition of handwritten numerals for Indian scripts is still a challenging task and there is spurt for work to be done in this area. A brief review of work done in recognition of handwritten numerals written in Devanagari script is given below.

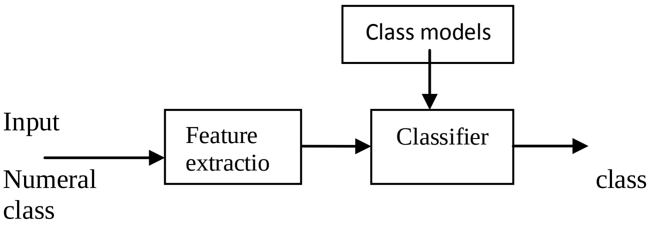
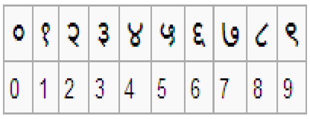


Figure1. Conventional HNR system

II. DATA SET AND PREPROCESSING

Devanagari script, originally developed to write Sanskrit, has descended from the Brahmi script sometime around the 11th century AD. It is adapted to write many Indic languages like Marathi, Mundari, Nepali, Konkani, Hindi and Sanskrit itself. Marathi is an Indo-Aryan language spoken by about 71 million people mainly in the Indian state of Maharashtra and neighboring states. Marathi is also spoken in Israel and Mauritius. Marathi is thought to be a descendent of Maharashtri, one of the Prakrit languages, which developed from Sanskrit. Since 1950 Marathi has been written with the Devanagari alphabet. Figure 2 below presents a listing of the symbols used in Marathi for the numbers from zero to nine.



०	१	२	३	४	५	६	७	८	९
0	1	2	3	4	5	6	7	8	9

Figure 2. Numerals 0 to 9 in Devanagari script

To the best of our knowledge standard dataset for Marathi numeral is not available till today. Therefore, dataset of Marathi handwritten numerals 0 to 9 is created by collecting the handwritten documents from writers. Data collection is done on a sheet specially designed for data collection. Writers from different professions were chosen including students, clerks, teachers, and vendors and were asked to write the numerals. No constraints were imposed on the use of ink or pen except that they have to write the numerals in the boxes of the sheets provided to them. A sample sheet of handwritten numerals is shown in figure 3.

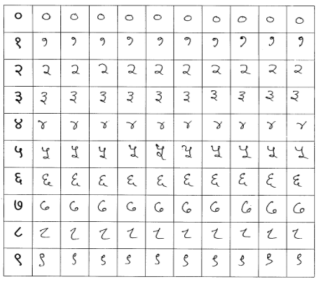


Figure 3. Sample sheet of handwritten numerals

III. FEATURE EXTRACTION

A well-defined feature extraction algorithm makes the classification process more effective and efficient. Two well-defined methods of feature extraction used in our method are Fourier descriptors and normalized chain codes.

Feature extraction stage in character recognition, as in any pattern recognition task, playsa major role in improving the recognition accuracy. The features are extracted from binary characters. Thus, the characteristics used for classification lie solely in the shape variations. Many characters are similar in shape or slight variation in the writing style may result into misclassification. The features selected should tackle all these problems. Moreover, a single feature extraction and classification stage may recognize a character which may not be recognized by the other feature extractor and classifier combination. Hence a hybrid system is needed that can recognize the characters over a wide range of varying conditions. The cropped characters in each structural class are resized to a fixed size before extracting the features. Three different features are extracted to be applied to three separate neural networks. The features extracted are pixel density features, Euclidean distance features and modified approximation wavelet features. The features extraction procedure is explained next.

Fourier Descriptors

Fourier transformation is widely used for shape analysis. 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 (xk, yk). Therefore, we can describe the contour as two parametric equations:

$$x(k) = xk$$

$$y(k) = yk$$

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

$$s(k) = x(k) + Iy(k)$$

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

The discrete Fourier transform of s(k) is

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

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,

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

The algorithm for computing Fourier descriptors is given below.

Input: Gray scale handwritten numeral image

Output: 32 dimensional Fourier descriptors

Method 1:

1. Remove noise by applying median filter three times.
2. Binarize the image by applying Otsu’s algorithm to obtain the binary image with numeral representing binary 1 and background 0
3. Remove noise (eg. Spikes) by applying erode and dilate morphological operations respectively.
4. To bring uniformity among the numerals, fit a bounding box to the numeral image, crop, and resize it to a size of 40 x 40 pixels.
5. Extract the boundary of the numeral image and resample the boundary in order to obtain a uniform resampling along the running arc length of the boundary.
6. Represent the boundary in the complex plane where the column-coordinate is the real part and the row-coordinate the imaginary part.
7. Compute Fourier transform and obtain the invariant 32 dimensional Fourier descriptors

The invariance is obtained by -nullifying the 0-th Fourier descriptor (position invariance), dividing all Fourier descriptors by the magnitude of the 1-st Fourier descriptor (size invariance) and only considering the magnitude of the Fourier descriptors (orientation and starting point invariance). By applying this normalization, the 0-th and 1-th Fourier descriptors do not provide any information. Hence these are eliminated. Experimentally we have chosen number of coefficients of Fourier descriptors to be 32. These descriptors, invariant to scale, translation and rotation, form the feature vector.

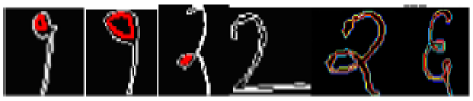


Figure 4. Contour of handwritten numerals

Median Filter

In signal processing, it is often desirable to be able to perform some kind of noise reduction on an image or signal. The **median filter** is a nonlinear digital filtering technique, often used to remove noise. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise .

Image Binarization

In computer vision and image processing, **Otsu's method** is used to automatically perform histogram shape-based image thresholding or, the reduction of a graylevel image to a binary image.

Morphological operations

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, you can construct a morphological operation that is sensitive to specific shapes in the input image.The most basic morphological operations are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the *structuring element* used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image. The rule used to process the pixels defines the operation as a dilation or an erosion. This table lists the rules for both dilation and erosion.

IV. CHAIN CODE

Chain codes are one of the shape representations which are used to represent a boundary by a connected sequence of straight line segments of specified length and direction.

This representation is based on 4-connectivity or 8-connectivity of the segments . The direction of each segment is coded by using a numbering scheme as shown in Figure 2 below. Chain codes based from this scheme are known as Freeman chain codes.

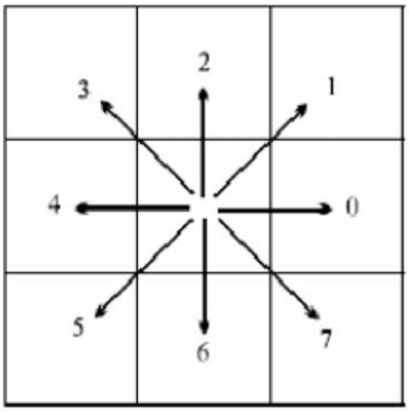


Figure 5: 8 directional neighborhoods.

Direction numbers for

(a) 4-directional chain codes,
(b) 8-directional chain code

A coding scheme for line structure must satisfy three objectives:

a. It must faithfully preserve the information of interest;

b. It must permit compact storage and convenient for display; and

c. It must facilitate any required processing.

According to, chain codes are a linear structure that results from quantization of the trajectory traced by the centers of adjacent boundary elements in an image array. A chain code can be generated by following a boundary of an object in a clockwise direction and assigning a direction to the segments connecting every pair of pixels. First, we pick a starting pixel location anywhere on the object boundary. Our aim is to find the next pixel in the boundary. There must be an adjoining boundary pixel at one of the eight locations surrounding the current boundary pixel. By looking at each of the eight adjoining pixels, we will find at least one that is also a boundary pixel. Depending on which one it is, we assign a numeric code of between 0 and 7 as already shown in Figure 6. For example, if the pixel found is located at the right of the current location or pixel, a code "0" is assigned. If the pixel found is directly to the upper right, a code "1" is assigned. The process of locating the next boundary pixel and assigning a code is repeated until we came back to our first location or boundary pixel. The result is a list of chain codes showing the direction taken in moving from each boundary pixel to the next. The process of finding the boundary pixel and assigning a code is shown in Figure 6.

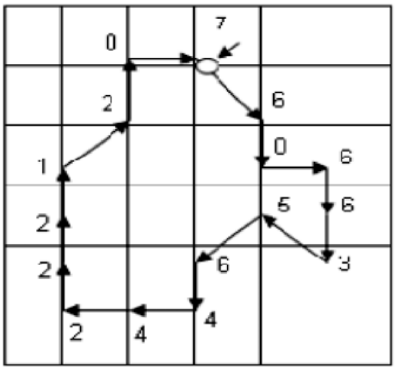


Figure 6: Direction and number assign of the shape

This method is unacceptable for two main reasons:
a. The resulting chain of codes tends to be quite long
b. Any small disturbances along the boundary due to noise or imperfect segmentation cause changes in the code that may not be related to the shape of the boundary.

Chain codes have been claimed as one of the techniques that are able to recognize characters and digits successfully. This is because of several advantages possessed by this technique as listed by. The first advantage over the representation of a binary object is that the chain codes are a compact representation of a binary object. Second, the chain codes are a translation invariant representation of a binary object. Due to that, it is easier to compare objects using this technique. The third advantage is that the chain code is a complete representation of an object or curve. This means that we can compute any shape feature from the chain codes. According to , chain codes provide a lossless compressing and preserving all topological and morphological information which bring out another benefit in terms of speed and effectiveness for the analysis of line patterns.

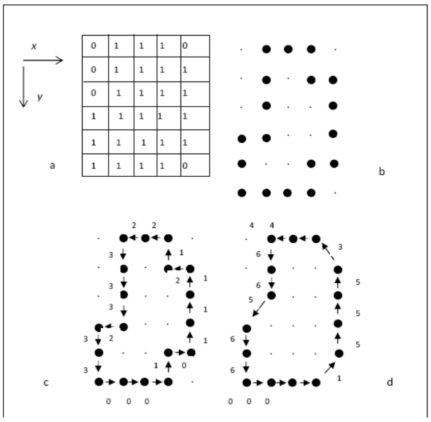
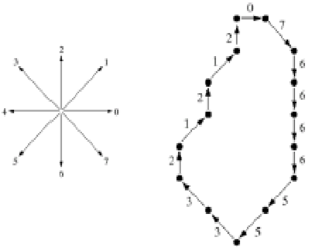


Fig. 7a & b) A 4-connected object and its boundary; c & d) Obtaining the chain code from the object in (a & b) with (c) for 4-connected and (d)

Data Freeman chain codes are generated by locating a boundary pixel, also called as starting pixel, and then moving along the boundary either clockwise or anticlockwise, finding next boundary pixel and assign this new pixel a code depending upon its location from the previous pixel. The process of finding the next pixel is terminated when starting pixel is encountered. The codes may be 4-directional or 8-directional depending upon 4connectivity or 8-connectivity of a pixel to its neighboring contour pixel. An 8-directional chain coded image is given in figure 8.



Chain code: 076666553321212

Figure 8: 8-connectivity chain code

The algorithm for generating the chain code is given below.

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.

It is observed that the chain code for different numerals has different length code and length of each chain code depends on the size of the handwritten numeral. More ever length of chain code is very high in case of certain handwritten numeral. We have solved this problem by normalizing the chain code values .

Normalized Chain Code

Assume that the following chain code is generated for a numeral contour by traversing it in anticlockwise (clockwise) direction.

V1=[1 2 2 2 1 2 1 2 1 1 2 0 1 0 1 0 0 0 0 0 0 7

7 0 0 0 1 0 0 0 0 0 0 0 0 07 0 0 7 0 7 6 6

6 7 6 6 7 6 6 6 6 6 7 6 6 6 6 5 6 6 6

5 6 6 5 5 6 5 4 4 5 4 5 4 5 4 4 4 5 4 4

4 4 4 3 4 4 4 3 4 4 3 4 3 4 4 3 2 3 3 2 3

2 3 2 3 2 2 2 2 2 2 2 2 2 2]

Compute the frequency of the codes 0,1,2…..7. For vector V1 we have the frequency vector V2 as below.

V2= [23 8 19 10 20 9 20 8]

The normalized frequency, represented by vector V3, is computed using the formula

$$V3 = \frac{V2}{|V1|} \quad \text{where } |V1| = \sum V2$$

For the example considered above, we have

V3= [0.1966 0.0684 0.1624 0.0855 0.0015 0.0769 0.1709 0.0684]

Finally, concatenating the vectors V2 and V3 we get the required feature vector of size 16.

The algorithm for computing normalized chain codes is given below.

Input: Gray scale handwritten numeral image

Output: Normalized chain code of length 16.

Method 2:

1. Perform steps 1 to 5 of method1.
2. Trace the boundary in counterclockwise direction and generate 8 dimensional chain codes 0 to 7
3. Compute the frequency of the codes 0 to 7.
4. Divide frequency of each code by sum of the frequencies.
5. Combine values obtained in step 3 and 4 to obtain feature vector of length 16.

V. SUPPORT VECTOR MACHINE

Support Vector Machine is supervised Machine Learning technique. The existence of SVM is shown in fig 8. *Computer Vision* is the broad area whereas *Machine Learning* is one of the application domains of Artificial Intelligence along with pattern recognition, Robotics, Natural Language Processing. Supervised learning, Un-supervised learning, Semi-supervised learning and reinforcement learning are various types of Machine Learning. Support Vector Machine (SVM) was first heard in 1992, introduced by Boser, Guyon, and Vapnik in COLT-92. Support vector machines (SVMs) are a set of related supervised learning methods used for classification and regression. They belong to a family of generalized linear classifiers.

In another terms, Support Vector Machine (SVM) is a classification and regression prediction tool that uses Machine Learning theory to maximize predictive accuracy while automatically avoiding over-fit to the data. Support Vector machines can be defined as systems which use hypothesis space of a linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory. The foundations of Support Vector Machines (SVM) have been developed by Vapnik and gained popularity due to many promising features such as better empirical performance. The formulation uses the Structural Risk Minimization (SRM) principle, which has been shown to be superior, to traditional Empirical Risk Minimization (ERM) principle, used by conventional neural networks. SRM minimizes an upper bound on the expected risk, where as ERM minimizes the error on the training data. It is this difference which equips SVM with a greater ability to generalize, which is the goal in statistical learning. SVMs were developed to solve the classification problem.

The objective of recognition is to interpret a sequence of numerals taken from the test set. The architecture of proposed system is given in fig. 8. The SVM (binary classifier) is applied to multiclass numeral recognition problem by using one-versus-rest type method. The SVM is trained with the training samples using linear kernel. Classifier performs its function in two phases; Training and Testing. After preprocessing and Feature Extraction process, Training is performed by considering the feature vectors which are stored in the form of matrices. Result of training is used for testing the numerals.

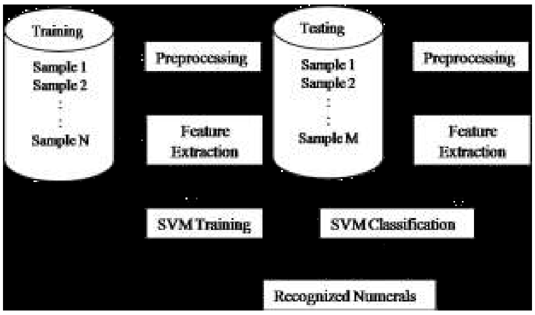


Figure 8:Architecture of SVM

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.

Algorithm : SVM Training phase
Input: Image, x of digit, i from dataset of n images.
Output: Feature vector f[ ] of dimension 18.
Method:
$d = 0\ldots 9$. $n = 1\ldots 2500$

```
For i Є d do
    For x Є n do
        Read image x of digit i.
        Get imgc by complementing pixel values of image x.
        Get amgt by performing thinning on imgc.
        Extract 7 features a[imgc] by using MI.
        Extract 7 features b[amgt] by using MI.
        Extract 4 features c[amgt] by using AMI.
        Merge all features in one feature vector, f[ ].
        Store f[ ] in training set, tr.
    end.
end.
Invoke training set, tr and group set, gs.
Perform training on tr and gs.
Return feature vector, f[ ] of dimension 18.
```

Algorithm : Recognition Phase
Input**:** Image *x* of Devanagari Handwritten Digit *i* from testing
dataset, *TS*. Feature vector, *f[ ]* of dimension 18.
Output**:** Recognition of digit *i*.
Method:

```
For i Є TS do
    Extract 18 features of x of i in one feature vector ft[ ].
    Compare ft[ ] with feature vector f[ ] generated in
    training phase.
    If match is found then
        Image x is recognized as digit i.
    else
        Image x does not recognize as digit i.
    end if.
end.
```

Support Vector Machines (SVM) has been considered as one of the powerful classifiers for character and numeral recognition. 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 . 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 and b have been determined by solving a quadratic problem.

VI. PROPOSED SYSTEM

Experiments were carried out on a database of 12690 isolated Marathi handwritten images obtained as described in section 2. The numeral images were grouped into five subsets to carry out five-fold cross validation. During each iteration, one subset was chosen as test set and rest of the subsets were combined to form training set. Fourier Descriptors and normalized chain codes are computed for all the images in the training set by performing method1 and method 2 described in section 3. The result of the training step consists of the (Model) set of support vectors determined by the SVM based method (Fig. 8).

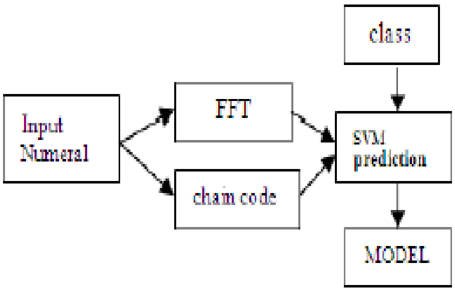


Figure 9. Training process

During the recognition step, the Fourier Descriptors and normalized chain codes are computed in the same way, and the model determined during the training step is used to perform the Decision. The output is the image class (Fig. 9).

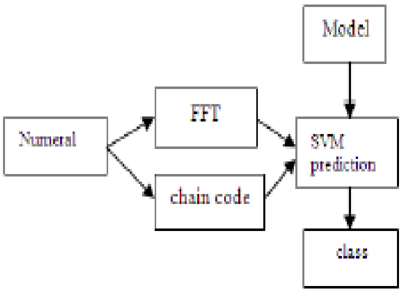


Figure 10. Recognition process

VII. CONCLUSION

In this system we have presented an efficient method for recognition of Marathi handwritten numerals using Fourier descriptors and normalized chain codes.

SVM is used for classification. The average recognition of 98.15 % is achieved using 5-fold cross validation method.

The proposed method is compared with other methods in terms of size of feature vector and recognition rate and is found to be very good. The proposed method can be extended to recognition of numerals of other Indic scripts.

REFERENCES

[1] Marathi Handwritten Numeral Recognition using Fourier Descriptors and Normalized Chain Code by G. G. Rajput,Department of Computer Science Gulbarga University, Gulbarga & S. M. Mali Department of Computer Science, Solapur University, Solapur

[2] V.N. Vapnik 1995, The Nature of Statistical Learning Theory, Springer, New York.

[3] R.O. Duda, P.E. Hart, D.G. Stork 2001. Pattern Classification, second ed., Wiley-Interscience, New York.

[4] T. Hastie, R. Tibshirani, J. Friedman 2001. The elements of statistical learning, Springer Series in Statistics, Springer, New York.

[5] Plamondon Rejean & Sargur N, Srihari 2000. On line & offline handwriting recognition: A comprehensive survey", IEEE Transactions on PAMI, 22 (1).

[6] U. Pal and B. B. Chaudhuri 2000. Automatic Recognition of Unconstrained Off-line Bangla Hand-written Numerals", Proc. Advances in Multimodal Interfaces, Springer Verlag Lecture Notes on Computer Science (LNCS-1948), 371378.

[7] N. Tripathy, M. Panda and U. Pal 2004. A System for Oriya Handwritten Numeral Recognition, SPIE Proceedings, Vol.-5296, Eds. E. H. Barney Smith, J. Hu and J. Allan, 174181.

[8] Y. Wen, Y. Lu, P. Shi 2007. Handwritten Bangla numeral recognition system and its application to postal automation, Pattern Recognition, 40 (1): 99-107.

[9] G.G. Rajput and Mallikarjun Hangarge 2007. Recognition of isolated handwritten Kannada numerals based on image fusion method', PReMI07, LNCS.4815,153-160.

[10] Reena Bajaj, Lipika Dey and Santanu Chaudhuri 2002. Marathi numeral recognition by combining decision of multiple connectionist classifiers, Vol. 27, Part 1,59–72, February 2002.