A STUDY OF EFFECT OF VARIOUS FACTORS ON RECOGNITION

Satish Kumar

Panjab University SSG Regional Centre, Hoshiarpur, Punjab (India)

ABSTRACT

The size of the training dataset plays an important role in handwritten recognition. Although various effective feature extraction methods have been proposed, even then there is a need of big training data set. In this paper, we have conducted experiments to study the effect of size of data set on recognition performance using chain code based feature. The effect of the character size normalization and increase in number of zones in chain code based feature are investigated. The effect of the headline removal on recognition performance is also investigated.

A classifier may work in single stage or in two stages. The effect on the recognition rate by designing a classifier in single stage and two stages is also examined. The classification method used is Multilayer perceptron. A self-designed database of Devanagari Handwritten Characters, collected from different persons, has been used for performing experiments. The strategy used to collect handwritten Devanagari character samples is also briefed.

Keywords: Handwritten, Devanagari, Chain code, Sample Size, Normalization, Headline.

I. INTRODUCTION

Handwriting is one among the natural ways of communication. Handwritten recognition is processes in which graphically hand printed scripts of a language are transformed into their symbolic representation. The handwritten expressions may be characters, words or lines pertaining to a particular language. They may be of any size, shape, orientation and style. In offline recognition the scripts are written on a document page and then scanned using a scanner and recognized in various stages. A survey on the handwritten recognition has been carried by Plamondon et al [1], Koerich et al [2] and Arica et al [3]. The review on work done for character recognition before 1990 is reported by Govindan et al [7].

Due to writing styles of different persons, there may be a lot of variations in size of different samples used for training and testing the system. It is essential to bring all the characters to some standard size before performing recognition. The various size normalization techniques have been investigated by Liu et al[10].

Chaincode representation of an object contours has been used by Madhvanath et al[13] and Kimura et al[21] for word recognition. Its performance on handwritten numerals has been studied by Liu et al[10], Hoque et al [14] and Cao et al[12]. It has powerful discrimination ability.

The artificial neural work based classifier have been studied and used for character recognition by Stefen et al [17], Amin et al [18], Cho [19] and many more. The detail survey about neural network based recognition methods has been given by Zhang[15]. A tutorial on artificial neural network is given by Jain et al[16].

The detail survey about the work done for most of the Indian languages script recognition is made by Pal et al [5]. The work on machine printed Devanagari has been made by Bansal et al [6,8], Pal et al [23] and Chaudhuri et al [9]. The work on handwritten Devanagari numeral is carried by Bajaj et al[22] and online numeral is done by connel et al[20]. The work on handwritten Devanagari character recognition is on its preliminary stage.

As such pattern recognition model to hand writing recognition consists of training the machines with handwritten samples of different persons. A lot of training and test samples are required. Bhattacharya et al [4] has designed a database of Devanagari handwritten numerals to carry research in this field. As far as it is concerned about handwritten Devanagari character recognition, there does hardly exist any database, as per my knowledge, either for isolated characters or isolated words. Due to lack of database a very little work has been done on handwritten Devanagari character recognition. There may be some work but that is on preliminary stage and has been performed on limited training and test data set. In order to carry our research on this topic we have created our own training and test character data set by collecting samples from different writers. It consists of about 26000 characters of 43 Devanagari basic characters.

The experiments reported here involve only the feature extraction and classification of the pre-segmented characters using chain code histograms. We are studying the effect of the training set size, number of regions taken and size normalization on the recognition rate using chain code based feature. The experiments have been performed on large number of training and test samples. Actually we are in search of a best feature for our application. We have also experimented with some other features but here we are discussing only chain code based feature.

II. HAND-PRINTED SAMPLE COLLECTION

There are number of methods of recognition has been given in literature. Some of these works are on isolated characters, some on words and others work on sentence level. The work done on sentence level is however negligible. The data set is prepared keeping in view the recognition methodology to be used for recognition purpose. Our sample set is of isolated handwritten Devanagari characters and consists of 43 basic characters.

To do this we have designed a handwritten Devanagari sample form. The form consists the entry space for all the middle zone full characters. In order to take more variable samples, a paragraph of words has been incorporated in the form. This paragraph consists of all 43 characters and 8 matras and some conjuncts. The paragraph is designed in such a way that a writer is able to read and write a full line in flow. The forms were given to different school and college level students and employees working in offices. The black sketch (narrow tip) and ball pens were used for filling the form. All images are scanned in black and white mode at 300 dpi. Each character written inside the box or paragraph retrieved manually and stored in a bitmapped file whose size is same as that of the character. The noise contained, if any, either due to writing instrument or scanning is removed. Some samples of our data set are given in Fig. 1. Due to space limitation, the sample form and database statistics is not described here.

III. CHAIN CODE REPRESENTATION OF CONTOUR

The contour of an image is the set of pixels; where each pixel has at least two black neighbors; that lie on the image boundary. In another way if a black pixel has at least a white pixel in its neighborhood then black pixel is regarded as on boundary. Locating contour of an image is not difficult task. Any contour present in an image can be located using Freeman chain code [11].

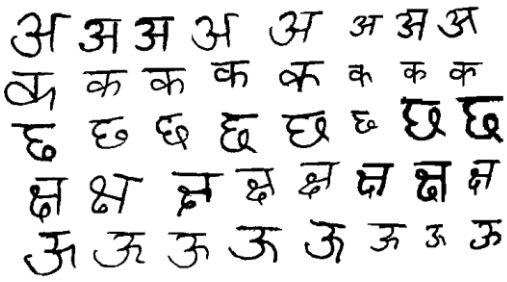

Fig. 1: Some samples from sample set.

Chain codes are a kind of directional codes. They represent a contour of an image using a sequence of codes where each code stands for the direction of a particular contour pixel w.r.t its adjoining contour pixels. The codes may be 4-directional or 8-directional depending upon 4-connectivity or 8-connectivity of a pixel to its neighboring contour pixel. The 8-directional codes corresponding to 8-neighbour pixels are given in Fig. 2.

To obtain chain codes, the image is scanned from top to bottom and left to right until a black boundary pixel is obtained. This is starting pixel and all the pixels present in this contour are located by tracking boundary sequentially along the contour and each boundary pixel is assigned a code using freeman algorithm.

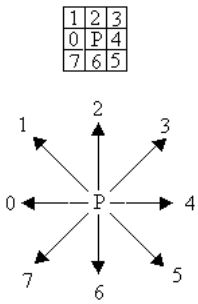

Fig. 2. 8-directional chain codes corresponding to 8-connectivity.

One of the advantages of the chain code representation of an image is that it can be fully recovered from its chain codes. Tracing the chain code of a contour provides the directions of its boundary pixel. The disadvantage associated with this is that it is difficult to extract contour of an image in noisy environment and a noise elimination algorithm is required before extracting contours.

IV. HEADER LINE REMOVAL AND SIZE NORMALIZATION

There is a horizontal line present in Devanagari characters at the top called as header line. The presence of header line effects the recognition rate a lot. So it is essential to remove this line. The horizontal projection histograms are used to remove this line. In horizontal histogram of a Devanagari character, there are peaks in the histograms where header line is present. The horizontal histograms of the upper 15% portion of the character are

computed. The position L as shown in Fig. 3, where this histogram maximizing, is located. The bottom position B of the header line is the location where the difference between maximum peak of histogram and the minimum valley of the histogram maximizes. The entire foreground pixels between T and B are removed

Fig. 3. Devanagari character 'th' before and after header line removal.

The size normalization of an image plays an important role. Extracting features from characters without normalizing not only poses some problems in feature representation but also degrades the performance of the recognition system. Size normalization is used to reduce the variations in size. The character image is enhanced /reduced to bring it to some standard size and this process is called as scaling. The size normalization method used here is, which preserves the aspect ratio, studied by Liu[10].

V. FEATURE EXTRACTION

There may be more than one contour present in an image. All possible existing contours are traced and their chain codes are produced. For our experiments, we have extracted 8-directional codes but these are normalized to 4-dirctional codes. The contour pixels having code 4 , 5, 6 and 7 are assigned 0, 1, 2 and 3 codes. The character bitmap is normalized to $N \times M$ pixels size, depending upon the regions taken for a given experiment, using normalization method referenced in paragraph 4 of this paper. In order to reduce the dimensionality of the feature vector, the character bitmap is divided into $n \times m$ equal size regions. The histograms of each chain code in each region is computed and normalized by the total number of pixels in each region. Since we have normalized 8-directional codes to 4-directional codes, the size of the feature vector is $4 \times n \times m$.

VI. CLASSIFICATION

In the classification stage the features extracted from an unknown character are compared with the features of the known characters (either stored in computer memory or used to trained the machine). The algorithm, that performs this task, is called as classifier. We have used MLP (Multilayer Percepron) for classification purpose.

6.1 Multilayer Feed Forward Network with Back Propagation

This is a kind of neural network in which the input signal flows through the network in forward direction through connection weights and error is back propagated and weights are updated\corrected. Actually, the error back-propagation algorithm uses gradient-descent method to minimize the squared error cost function. Where weight adjustment between any neuron *j* and *k* is proportional to the negative gradient of the error, generated at *k*, with respect to weight

$$\Delta w_{jk}(t) = -\eta \frac{\partial E(t)}{\partial w_{jk}} \qquad (1)$$

where $E(t)$ is squared error cost function. The equation (1) can be expressed as:

$$\Delta w_{jk}(t) = \eta \delta_k(t) y_j(t) \quad (2)$$

where δ_k (t) is gradient of error, $y_j(t)$ is input signal to k^{th} neuron from j^{th} neuron, t represents t^{th} iteration and η is learning parameter. One more constant term α called as momentum term was added to change in weights The effect of adding momentum term is to keep the weights changing and do not struck at local minima. Where $0 \le \alpha < 1$.

$$\Delta w_{jk}(t) = \alpha \Delta w_{jk}(t-1) + \eta y_j(t) y`_k(t) e_k(t) \quad (3)$$

The weight available for next iterations is

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk}(t) \quad (4)$$

VII. EXPERIMENTAL RESULTS

We have conducted various experiments, taking different size of training set, different size of characters and number of regions. The experiments are conducted for 43 basic middle zone full box Devanagari characters which also includes a 'kanna'. In this paper we have used 600 characters per class out of which 150 characters/class were used for testing the results and are not used for training purpose. All the characters have been trained in a single network. The size of the input layer is the size of the feature vector and the size of the output layer is 43. The network consists of a single hidden layer having 130 nodes.

7.1 The Effect of Training Sample Set Size

To conduct this experiment, the character bitmap is normalized to 50×50 pixels size and it is divided into 5×5 regions. The histogram for each chain code in each region is computed and normalized with the total number of pixels in each region. This gives a feature vector of size 4×5×5. The two-stage classifier is used. Table 1 gives the results. The results have been tested on 150 characters/class which have not been used for training. The analysis of results is given in Fig. 4.

7.2 The Effect of the Size Normalization

In order to investigate the effect of size normalization, we have conducted experiments using both normalized and un normalized characters. The dataset of each character is partitioned into 4 subsets A, B, C and D. In each experiment, three sets were used to train the classifier and fourth set was used for testing purpose. The results against set A mean trained on B, C, D and tested on subset A. The single stage classifier is used. The table 2 gives the experimental results in percent. The analysis of results is given in Fig. 5.

TABLE 1. Recognition results with different size of training set.

Sr. No	Number of Training samples per classes	Results (Percent)
1	50	78.7
2	150	84.7
3	250	87.3
4	350	89.1
5	450	90.2

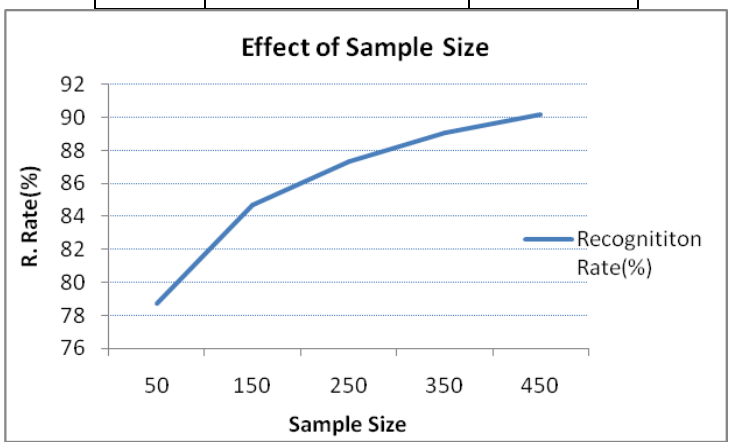

Fig. 4: Result Analysis Size of Training Dataset

TABLE 2. Recognition results with normalized and un-normalized data set.

Un-Normalize	A	76.4
	B	71.4
	C	70.3
	D	75.9
	Avg.	73.5
Normalized	A	90.2
	B	85.9
	C	85.7
	D	90.4
	Avg.	88.0

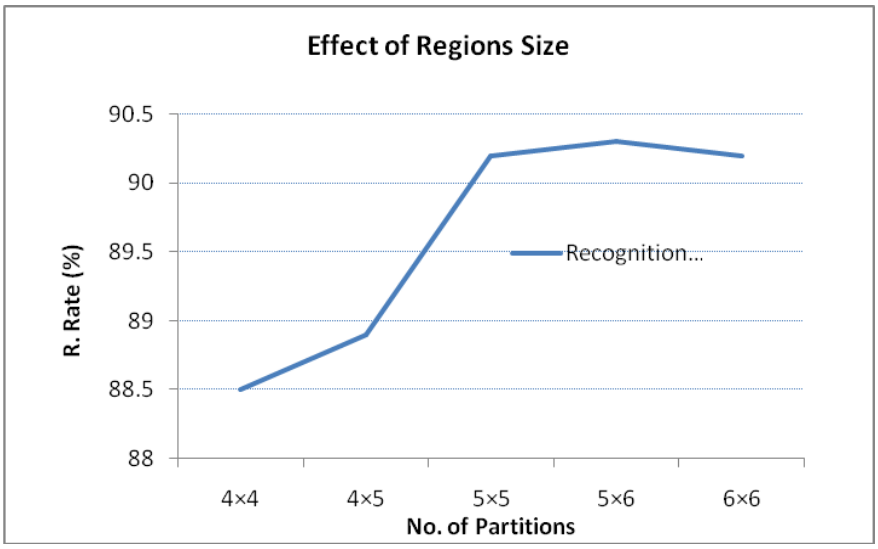

Fig. 5: Result Analysis Size of No. of Region

7.3 The Effect of the Number of Regions

For this particular feature vector the character bitmap is divided into number of regions, in order to reduce the size of feature vector. To know the effect of the number of regions, we have conducted experiments partitioning the bitmap in number of regions. The size of the character bitmap is taken in such a way that the size of each region is same. For each kind of experiment here we have used B, C and D subset for training and A subset for testing. The table 3 gives the experimental results. The analysis of results is given in Fig. 6.

TABLE 3. Recognition results with different number of regions taken.

Sr. No.	Character Size (pixels)	Number of Regions	Feature vector size	Recognition rate(Percent)
1	48×48	4×4	64	88.5
2	48×50	4×5	80	88.9
3	50×50	5×5	100	90.2
4	50×48	5×6	120	90.3
5	48×48	6×6	144	90.2

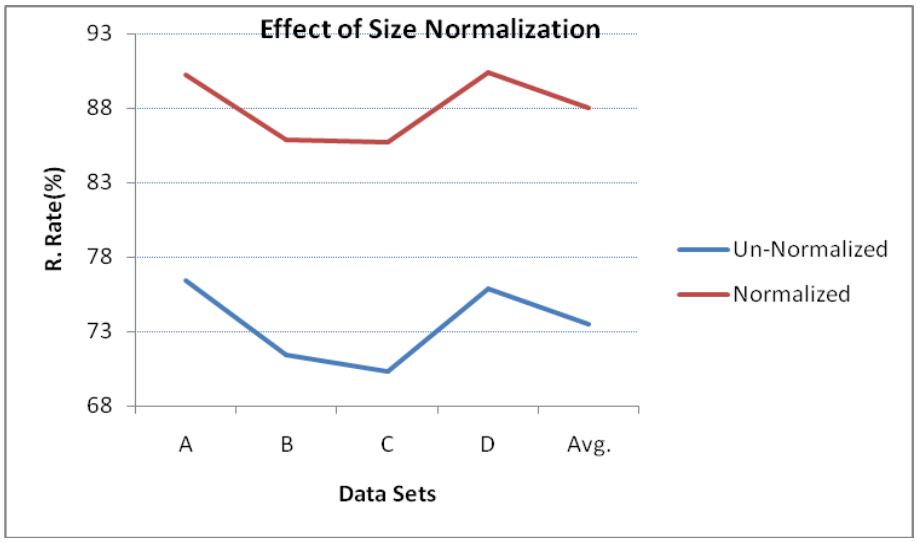

Fig. 6: Result Analysis Effect of Size Normalization.

7.4 The Effect of Headline Removal

In order to investigate the effect of removal of head-line from Devanagari Hand-printed characters, we have conducted experiments with headline as well without headline. The strategy used for conducting the experiments is same as discussed in sub-section 7.2 for normalized characters images. The details of the experiments are given in Table 4. The analysis of results is given in Fig. 7.

TABLE 4. Recognition results with & without Headline.

With Headline	A	74.3
	B	72.1
	C	69.7
	D	71.2
	Avg.	71.8
Without Headline	A	90.2
	B	85.9
	C	85.7
	D	90.4
	Avg.	88.0

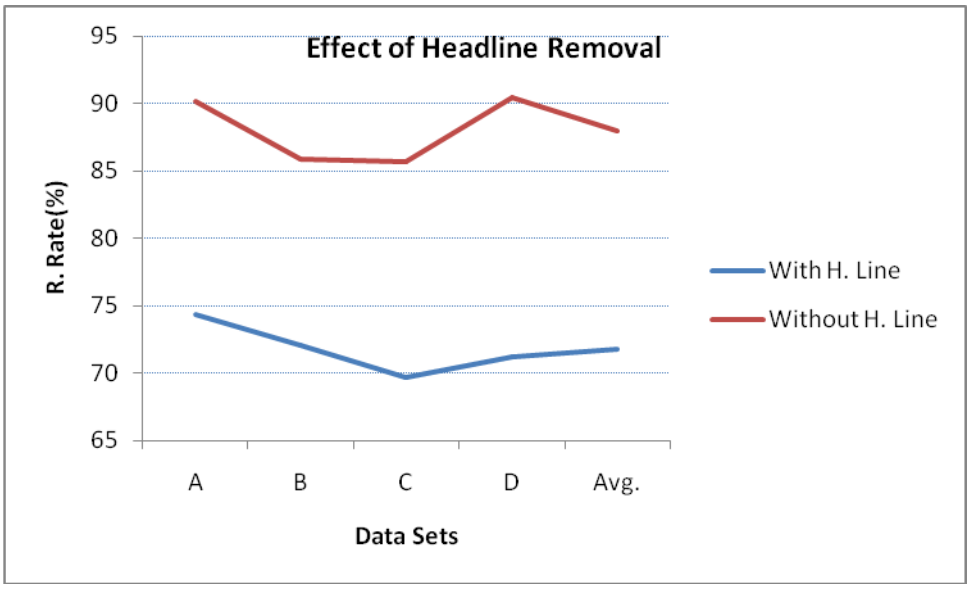

Fig. 7: Result Analysis Effect of Headline Removal.

VIII. CONCLUSION

In this we have studied the effect of various factors on the recognition rate using large dataset of Devanagari handwritten characters. It is clear that with increase in training set size the recognition rate increases. This is due to the fact that small training data set does not cover all the variation in arising due to writing styles. In case of size normalization, the recognition rate is higher where normalized data set is used to train and test the network. The difference in recognition rates with normalized and un-normalized data is quite large. Upto first four rows in table 3, the recognition rate increases and there after the recognition rate is not effected much with increase in regions. This is due to the curse of dimensionality. As far as the effect of headline removal is concerned, the recognition rate with without line is higher as compared with headline. Therefore, it is essential to study the effect of various factors on recognition rate to get optimal results.

REFERENCES

[1] R. Plamondon and S. N. Srihari, "On-Line and Off-Line Handwritting Recognition : A Comprensive Survey", IEEE trans. on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, 2000, pp. 63-84.

[2] A.L. Koerich, R. Sabourin and C.Y. Suen, "Large Vocabulary Off-line Handwriting Recognition: A survey, Pattern Analysis Applications Vol. 6, 2003, pp. 97-121.

[3] N. Arica and T. Yarman-Vural, "An Overview of Character Recognition Focused on Off-Line Handwriting", IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, Vol. 31, No. 2, May 2001.

[4] U. Bhattacharya, B. B. Chaudhuri, "Database for Research On Recognition of Handwritten Characters of Indian scripts" Proceedings of Int. Conf. on Document Analysis and Recognition II , 2005, pp. 789-793.

[5] U. Pal, B.B. Chaudhuri, "Indian Script Character Recognition: A Survey", Pattern Recognition, Vol. 37, 2004, pp. 1887-1899.

[6] V. Bansal and R.M.K. Sinha, "Integreted Knowledge Sources in Devaganari Text Recognition", IEEE Trans. on System, Man, Cybernetics, Vol. 30, No. 4, 2000, pp. 500-505.

[7] V.K. Govindan and A.P.Shivaprasad, "Character Recognition: A Review", Pattern Recognition, Vol. 23, No. 7, 1990, pp. 671-683.

[8] V. Bansal and R. Sinha, "On How to Describe Shapes of Devanagari Characters and Use them for Recognition", Proceedings of International Conference on Document Analysis and Recognition, Banglore, India, Sept. 20-22, 1999, pp. 410-413.

[9] B.B. Chaudhuri and U. Pal, "An OCR System to Read Two Indian Language Scripts: Bangla and Devanagri", Proceedings of International Conference on Document Analysis and Recognition, 1997, pp. 1011-1015.

[10] C.-L. Liu, K. Nakashima, H. Sako and H. Fujisawa, "Handwritten Digit Recognition: Investigation of Normalization and Feature Extraction Techniques" , Pattern Recognition, Vol. 37, 2004, pp. 265-279.

[11] H. Freeman, "Computer processing of line drawing images", ACM Computing Surveys, Vol. 6, No. 1, 1974, pp. 57-98.

[12] J. Cao, M. Ahmadi and M. Shridhar, "Recognition of Handwritten Numerals with Multiple Feature and Multistage classifier", Pattern Recognition Vol. 28, No. 2, 1995, pp. 153-160.

[13] S. Madhvanath, G. Kim and V. Govindaraju, "Chaincode Contour Processing for Handwritten Word Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 9, Sepetember 1999, pp. 928-932.

[14] S. Hoque, K. Sirlantzis and M.C. Fairhurst , "A New Chaincode Quantization Approach Enabling High Performance Handwriting Recognition Based on Multiple-classifier Scheme", Proceedings of Seventh International Conference on Document Analysis and Recognition , 2003.

[15] G.P. Zhang, "Neural Networks for Classification: A Survey", IEEE Transactions on Systems, Man, and Cybernatics—part c: Applications and Reviews, Vol. 30, No. 4, November 2000.

[16] A. K. Jain, J. Mao, and K. Mohiuddin, "Artificial Neural Networks: A Tutorial" , IEEE Computer Special Issue on Neural Computing, March 1996.

[17] K. Stefen, P. Leon and D. Gerard, "Handwritten Digit Recognition by Neural Networks with Single-Layer Training", IEEE Transaction on Neural Networks, 1992.

[18] A. Amin and H. Al-Sadoun, "Handwritten Arabic Character Recognition System using An Artificial Neural Network", Pattern Recognition , Vol. 29, 1996, pp. 663-675.

[19] S.-B. Cho, "Neural-Network Classifiers for Recognizing Totally Unconstrained Handwritten Numerals", IEEE Trans. on Neural Networks, Vol. 8, No.1, January 1997.

[20] S. D. Connel, R.M.K. Sinha and Anil K. Jain, "Recognition of Unconstraint On-Line Devanagari Characters", Proceedings of the International Conference on Pattern Recognition, 2000.

[21] F. Kimura, M. Shridhar and Z. Chen, "Improvement of a Lexicon Directed Algorithm for Recognition of Unconstrained Handwritten Words, International Conference on Document Analysis and Recognition, 1993, pp. 18-22.

[22] R. Bajaj, L. Dey, S. Chaudhury, "Devanagari Numeral Recognition by Combining Decision of Multiple Connectionist Classifiers", Sadhna, Vol. 27, Part 1, Feb. 2002, pp. 59-72.

[23] U. Pal, B.B Chaudhuri, "Printed Devanagari Script OCR system", Vivek, Vol. 10, 1997, pp. 12-24.