

Handwritten Gurmukhi Numeral Recognition using Different Feature Sets

Kartar Singh Siddharth

Renu Dhir

Rajneesh Rani

Department of Computer Science and Engineering
Dr B R Ambedkar National Institute of Technology
Jalandhar- 144011, Punjab, India

ABSTRACT

Recently there is an emerging trend in the research to recognize handwritten characters and numerals of many Indian languages and scripts. In this manuscript we have practiced the recognition of handwritten Gurmukhi numerals. We have used three different feature sets. First feature set is comprised of distance profiles having 128 features. Second feature set is comprised of different types of projection histograms having 190 features. Third feature set is comprised of zonal density and Background Directional Distribution (BDD) forming 144 features. The SVM classifier with RBF (Radial Basis Function) kernel is used for classification. We have obtained the 5-fold cross validation accuracy as 99.2% using second feature set consisting of 190 projection histogram features. On third and first feature sets recognition rates 99.13% and 98% are observed. To obtain better results pre-processing of noise removal and normalization processes before feature extraction are recommended, which are also practiced in our approach.

General Terms

Pattern Recognition, OCR, Handwritten character recognition, Gurmukhi script.

Keywords

Handwritten Gurmukhi numeral recognition, Zonal density, Projection histogram, Distance Profiles, Background Directional Distribution (BDD), SVM classifier, RBF kernel..

1. INTRODUCTION

Recently, many research articles are available for recognition of handwritten characters and numerals for Devnagari among Indian scripts. However, the approaches for character recognition for Indian scripts other than Devnagari are also approached, but such practices are comparatively fewer.

G.S. Lehal and Nivedita Bhatt [1] have proposed a system to recognise both Devnagari and English handwritten numerals. They used a set of global and local features, which were derived from the right and left projection profiles of the numeral image. They tested their system on both Devnagari and English numerals independently and found that recognition rate for Devnagari numerals was better. For Devnagari numerals they found recognition rate of 89% and confusion rate of 4.5%. For English set they found recognition rate of 78.4% and confusion rate of 18%.

Reena Bajaj et al. [2] have used three different types of features namely density, moment and descriptive component features to recognize the Devnagari numerals. They also used three

different neural classifiers and finally the outputs of three classifiers were combined using a connectionist scheme. Thus they obtained classification rate of 89.68% by combining all classifier.

U Bhattacharya and B.B. Chaudhuri [3] have proposed majority voting scheme for multi-resolution recognition of hand-printed numerals. They used the features based on wavelet transforms at different resolution levels and multilayer perceptron for classification purpose. They achieved 97.16% recognition rate on a test set of 5000 Bangla numerals.

U. Bhattacharya et al. [4] have used ANN and HMM for recognition of handwritten Devnagari numerals. In their proposed scheme they obtained 92.83% recognition rate.

U. Pal et al. [5] have proposed a modified quadratic classifier based scheme towards the recognition of off-line handwritten numerals of six popular Indian scripts namely Devnagari, Bangla, Telugu, Oriya, Kannada and Tamil scripts for their experiment. The features used in the classifier were obtained from the directional information of the numerals. They obtained 99.56%, 98.99%, 99.37%, 98.40%, 98.71% and 98.51% accuracy for the scripts respectively.

Sontakke and Patil et al. [6] have proposed general fuzzy hyperline segment neural network to recognize handwritten Devnagari numerals. They proposed a rotation, scale and translation invariant algorithm and reported 99.5% recognition rate.

Ramteke and Mehrotra [7] have used a method based on invariant moment and the divisions of numeral image for recognition of handwritten Devnagari numerals. They adopted Gaussian Distribution Function for classification and achieved 92% success rate.

Shrivastava and Gharde [12] have used Moment Invariant and Affine Moment Invariant techniques for feature extraction and SVM classifier and reported 99.48% recognition rate for handwritten Devnagari Numerals.

Mahesh Jangid and Kartar Singh et al. [13] have used a feature extraction technique based on recursive subdivision of the character image to recognize handwritten Devnagari numerals. The character image was subdivided at all iterations such that the resulting sub-image had balanced number of foreground pixels as possible. They achieved 98.98% recognition rate using SVM classifier.

There are also some approaches practiced for recognition of numerals of Indian scripts other than Devnagari. Rajput and Mali [8] have recognised isolated Marathi numerals using Fourier descriptors. They used 64 dimensional Fourier descriptors representing the shape of the numerals and invariant to rotation, scale and translation. They used three different classifiers namely NN, KNN and SVM independently and achieved 97.05%, 97.04% and 97.85% accuracies respectively.

Apurva Desai [9] have used four different profiles of digits namely horizontal, vertical, and left and right diagonal profiles to recognize Gujarati Numerals. He used neural network and achieved 82% success rate.

Dharamveer Sharma et al. [10] first extracted Gurmukhi digits from Gurmukhi documents and then used structural and statistical features to recognize. They tested the technique on both Roman and Gurmukhi digits and reported recognition rates of 95% and 92.6% respectively.

Ubeeka Jain et al. [11] have practiced an approach based neocognitron to recognize Gurmukhi character set including Gurmukhi numerals. They achieved 92.78% recognition rate.

In the following sections dataset generation, pre-processing, and proposed methodology including feature extraction, result analysis and conclusion are discussed.

2. DATASET

The dataset of Gurmukhi numerals for our implementation is collected from 15 different persons. Each writer contributed to write 10 samples of each of numeral of 10 different Gurmukhi digits. These samples are taken on white papers written in an isolated manner.

The table 1 shows some of the samples of our collected dataset. These samples are transformed in gray image. Among these samples, some distortions and irregularities are also incorporated by writers.

TABLE 1: HANDWRITTEN SAMPLES OF GURMUKHI NUMERALS

Digit	Samples				
0					
1					
2					
3					
4					
5					
6					
7					

8					
9					

In pre-processing we applied many techniques like median filtration, dilation, isolated pixels removal and many other morphological operations to bridge unconnected pixels and to remove spur pixels etc. Before extracting the features we normalized the pre-processed numeral images to 32*32 pixel size.

3. FEATURE EXTRACTION

We have used following three sets of features extracted to recognize Gurmukhi numerals. These approaches are adopted from our earlier practice [14] used to recognize isolated Gurmukhi handwritten characters.

1. Distance Profile Features
2. Projection Histogram Features
3. Zonal density (ZD) and Background Directional Distribution (BDD) Features.

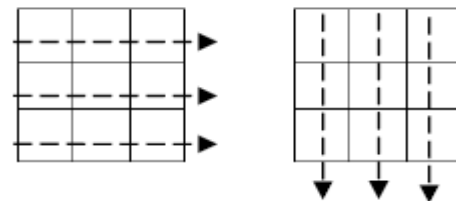
3.1 Distance Profile Features

In our approach we have used distance profiles using distance computation from bounding box to outer edges of character from four sides- two in horizontal direction from left and right sides and other two in vertical direction from top and bottom side.

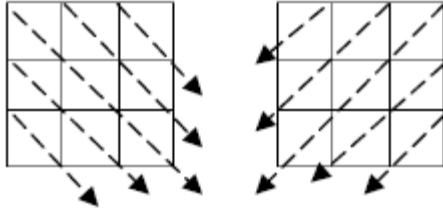
Left and right profiles are traced by horizontal traversing of distance from left bounding box in forward direction and from right bounding box in backward direction respectively to outer edges of character. Similarly, top and bottom profiles are traced by vertical traversing of distance from top bounding box in downward direction and from bottom bounding box in upward direction respectively to outer edges of character. The size of each profile in our approach is 32 similar to number of pixels in each row or column forming total 128 features by all four types of profiles.

3.2 Projection Histogram Features

Projection histograms count the number of foreground pixels in specified direction. In our approach we have used four directions of horizontal, vertical and both diagonal traversing.



(a) Horizontal Histogram (b) Vertical Histogram



(c) Diagonal-1 Histogram (d) Diagonal-2 Histogram

Fig. 1. Evaluation of 4 types of Projection Histograms on 3*3 patterns

Thus in our approach we have created four types of projection histograms: horizontal, vertical, diagonal-1 (left diagonal) and diagonal-2 (right diagonal). These projection histograms for a 3*3 pattern are depicted in figure 1.

In our approach projection histograms are computed by counting the number of foreground pixels. In horizontal histogram these pixels are counted by row wise i.e. for each pixel row. In vertical histogram the pixels are counted by column wise. In diagonal-1 histogram the pixels are counted by left diagonal wise. In diagonal-2 histogram the pixels are counted by right diagonal wise. The lengths of these features are 32, 32, 63 and 63 respectively according to lines of traversing forming total 190 features

3.3 Zonal Density and Background Directional Distribution Features

In zoning, the character image is divided into N×M zones. From each zone features are extracted to form the feature vector. The goal of zoning is to obtain the local characteristics instead of global characteristics. We have created 16 (4*4) zones of 8*8 pixel size each out of our 32*32 normalized samples by horizontal and vertical division. By dividing the number of foreground pixels in each zone by total number of pixels in each zone i.e. 64 we obtained the density of each zone. Thus we obtained 16 zoning density features.

We have considered the directional distribution of neighboring background pixels to foreground pixels. We computed 8 directional distribution features. To calculate directional distribution values of background pixels for each foreground pixel, we have used the masks for each direction shown in figure 2(b). The pixel at center 'X' is foreground pixel under consideration to calculate directional distribution values of background. The weight for each direction is computed by using specific mask in particular direction depicting cumulative fractions of background pixels in particular direction.

Algorithm for computation of BDD features:

// image size=32*32, number of zones =16 // (4*4), pixels in each zone=64(8*8)

(1) START

(2) Initialize D1,D2,... D8 = 0;

(3) Repeat step 4 for each zone Z(r,c)

//where, r=1,2,3,4; c=1,2,3,4;

(4) Repeat step 5 for each pixel P(m,n) of the zone Z(r,c)

//where, m=1,2,...8; n=1,2,...8;

(5) IF pixel P(m,n) is foreground

Then, 8 distribution features (d1,d2 ...d8) for zone Z(r,c) are computed by summing up the mask values specified for neighbouring background pixels in particular direction (see fig 2(b)) as follows:

$$D1 = D1 + 2 * \sim P(m, n + 1) + \sim P(m - 1, n + 1) + P(m + 1, n + 1);$$

$$D2 = D2 + 2 * \sim P(m - 1, n + 1) + \sim P(m - 1, n) + \sim P(m, n + 1);$$

$$D3 = D3 + 2 * \sim P(m - 1, n) + \sim P(m - 1, n - 1) + \sim P(m - 1, n + 1);$$

$$D4 = D4 + 2 * \sim P(m - 1, n - 1) + \sim P(m, n - 1) + \sim P(m - 1, n);$$

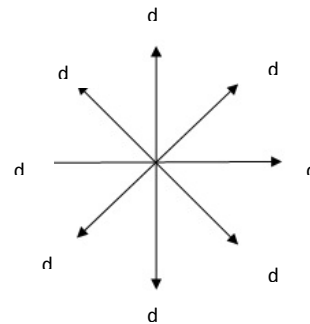
$$D5 = d5 + 2 * \sim P(m, n - 1) + \sim P(m - 1, n - 1) + \sim P(m + 1, n - 1);$$

$$D6 = d6 + 2 * \sim P(m + 1, n - 1) + \sim P(m, n - 1) + \sim P(m + 1, n);$$

$$D7 = d7 + 2 * \sim P(m + 1, n) + \sim P(m + 1, n - 1) + \sim P(m + 1, n + 1);$$

$$D8 = d8 + 2 * \sim P(m + 1, n + 1) + \sim P(m + 1, n) + \sim P(m, n + 1);$$

(6) STOP

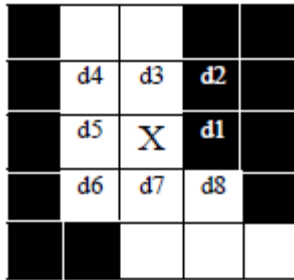


(a) 8 directions

0	0	1	0	1	2	1	2	1	2	1	0
0	X	2	0	X	1	0	X	0	1	X	0
0	0	1	0	0	0	0	0	0	0	0	0

1	0	0	0	0	0	0	0	0	0	0	0
2	X	0	1	X	0	0	X	0	0	X	1
1	0	0	2	1	0	1	2	1	0	1	2

(b) Masks used to compute different directional distributions



(c) An example of 3*3 sample

Fig. 2. Computation of Background Directional Features

Above algorithm describes the computation of BDD features. Eight directional features D1,D2, ... D8 are computed for each zone.

To illustrate the computation of BDD features, we have to compute directional distribution value for foreground pixel ‘X’ in direction d1 for the sample given in figure 2(c). We need to superimpose the mask for d1 direction on the sample image coinciding the centered pixel X. The sum of mask values of d1 mask coinciding to background pixels neighbouring to X in figure 2(c) i.e. d1 and d2 (i.e. 1+2) will be feature value in direction d1. Similarly we obtained all directional distribution values for each foreground pixel. Then, we summed up all similar directional distribution values for all pixels in each zone. Combining 16 zoning features we have 144 features (8*16 BDD + 16 density) computed in this set.

4. RECOGNITION RESULTS AND ANALYSIS

We have used SVM classifiers for recognition. Basically SVM classifies objects into binary classes but it can be extended to classify multiple classes. We have obtained such multiclass SVM tool LIBSVM available at [15]. We have used RBF (Radial Basis Function) kernel which is also common choice, in our recognition. RBF has single kernel parameter gamma (γ or γ). Additionally there is another parameter with SVM classifier called soft margin or penalty parameter (C).

4.1 5-Fold Cross Validation

In general, In V-fold cross validation we first divide the training set into V equal subsets. Then one subset is used to test by classifier trained by other remaining V-1 subsets. By cross validation each sample of train data is predicted and it gives the percentage of correctly recognized dataset.

In our approach we have used 5-fold cross validation to obtain recognition rate.

We first tested on small samples on all possible parameters giving optimized results and then by refinement on complete dataset we finally discovered the parameters’ combination giving optimum cross validation accuracy.

The table 2 depicts the optimized results obtained with different features set at refined parameters. The result variation is more sensitive to value γ comparison to C.

Feature Set	Recognition Rate	Parameters
Distance Profiles (128)	98%	C=16; $\gamma = 0.03- 0.15$
Projection Histograms (190)	99.2%	C=16; $\gamma = 0.05-0.15$
Zonal density and BDD (144)	99.13%	C=16; $\gamma = 0.05-0.55$

While observing the results at other values of parameter C, it is analysed that increasing the value of C irrespective of any change in γ slightly decreases the recognition rate, but after a certain increment normally after 64 at higher values of C the recognition rate becomes stable. In contrast, the recognition rate always changes with the change in γ . The optimized results are obtained at C=16 and γ value in range 2^{-5} to 2^{-1} .

Thus we can conclude that we have obtained the maximum recognition rate as 99.2% at 190 projection histogram features among all three feature sets. Secondly, we obtained 99.13% recognition rate at 144 zonal density and Background Directional Distribution features. Thirdly, we obtained 98% recognition rate at 128 distance profile features. It is clear here that feature set having large number of features is giving higher recognition rate.

But at the overall point of efficiency and performance, the recognition results using third feature set using zonal density and BDD features are best. It is because it has slight reduction in recognition rate (99.2% to 99.13%) but a significant reduction in number of features (190 to 144) comparative to second feature set. Hence it reduces computation complexity and hence time consumption while processing lesser number of features. In other way, comparing with first feature set, third features set provides significant increase in recognition rate (98% to 99.13%) while slight increase in number of features (128 to 144).

5. REFERENCES

- [1] G.S. Lehal, Nivedan Bhatt, “A Recognition System for Devnagari and English Handwritten Numerals”, *Proc. ICMI, Springer*, pp. 442-449, 2000.
- [2] Reena Bajaj, LipikaDey, ShantanuChaudhuri, “Devnagari numeral recognition by combining decision of multiple connectionist classifiers”, *Sadhana* Vol. 27, Part 1, pp. 59–72, February 2002.
- [3] U. Bhattacharya, B. B. Chaudhuri, "A Majority Voting Scheme for Multiresolution Recognition of Handprinted Numerals," *Seventh International Conference on Document Analysis and Recognition (ICDAR) - Volume 1*, pp. 16, 2003.
- [4] U. Bhattacharya, S.K. Parui, B. Shaw, K. Bhattacharya, “Neural Combination of ANN and HMM for Handwritten Devanagari Numeral Recognition”, *Tenth International Workshop on Frontiers in Handwriting Recognition*, 2006.

- [5] U. Pal, T. Wakabayashi, N. Sharma, F. Kimura, "Handwritten Numeral Recognition of Six Popular Indian Scripts," *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, Vol.2, pp.749-753, 2007.
- [6] P.M. Patil, T.R. Sontakke, "Rotation, scale and translation invariant handwritten Devanagari numeral character recognition using general fuzzy neural network", *Pattern Recognition, Elsevier*, Vol. 40, Issue 7, pp. 2110-2117, July 2007.
- [7] R.J. Ramteke, S.C. Mehrotra, "Recognition of Handwritten Devanagari Numerals", *International Journal of Computer Processing of Oriental Languages, Chinese Language Computer Society & World Scientific Publishing Company*, 2008.
- [8] G.G. Rajput, S.M. Mali, "Fourier Descriptor Based Isolated Marathi Handwritten Numeral Recognition", *International Journal of Computer Applications(IJCA)*, Vol. 3, No. 4, June 2010.
- [9] Apurva A. Desai, "Gujarati Handwritten Numeral Optical Character Reorganization through Neural Network", *Pattern Recognition*, Vol. 43, Issue 7, pp. 2582-89, July 2010
- [10] D. Sharma, G. S. Lehal, PreetyKathuria, "Digit Extraction and Recognition from Machine Printed Gurmukhi Documents", *MORC Spain*, 2009.
- [11] Ubeeka Jain, D. Sharma, "Recognition of Isolated Handwritten Characters of Gurumukhi Script using Neocognitron", *International Journal of Computer Applications (IJCA)*, Vol. 4, No. 8, 2010.
- [12] Shailedra Kumar Shrivastava, Sanjay S. Gharde, "Support Vector Machine for Handwritten Devanagari Numeral Recognition", *International Journal of Computer Applications (IJCA)*, Vol. 7, No. 11, October 2010
- [13] Mahesh Jangid, Kartar Singh, Renu Dhir, Rajneesh Rani, "Performance Comparison of Devanagari Handwritten Numerals Recognition", *International Journal of Computer Applications (IJCA)*, Vol. 22, No.1, May 2011.
- [14] Kartar Singh Siddharth, Mahesh Jangid, Renu Dhir, Rajneesh Rani, "Handwritten Gurmukhi Character Recognition Using Statistical and Background Directional Distribution Features", *International Journal of Computer Science and Engineering (IJCSE)*, Vol. 3, No. 6, June 2011.
- [15] Chih-Chung Chang and Chih-Jen Lin, LIBSVM: a library for support vector machines, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>