

Combination of Different Feature Sets and SVM Classifier for Handwritten Gurumukhi Numeral Recognition

Anita Rani

Rajneesh Rani

Renu Dhir

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

ABSTRACT

A lot of research has been done in recognizing handwritten characters in many languages like Chinese, Arabic, Devnagari, Urdu and English. This paper focuses on the problem of recognition of isolated handwritten numerals in Gurumukhi script. We have used different feature extraction techniques such as projection histograms, background directional distribution (BDD) and zone based diagonal features. Projection Histograms count the number of foreground pixels in different directions such as horizontal, vertical, left diagonal and right diagonal creating 190 features. In Background Directional Distribution (BDD) features background distribution of neighbouring background pixels to foreground pixels in 8-different directions is considered forming a total of 128 features. In the computation of diagonal features, image is divided into 64 equal zones each of size 4×4 pixels then features are extracted from the pixels of each zone by moving along its diagonal, thus consisting of total 64 features. Different combinations of these features are used for forming different feature vectors. These feature vectors are classified using SVM classifier as 5-fold cross validation with RBF (radial basis function) kernel. The highest accuracy achieved is 99.4% of whole database using combination of background directional distribution and diagonal features with SVM classifier.

General Terms

Pattern Recognition, OCR, Handwritten Character Recognition, Feature Extraction.

Keywords

Handwritten Gurumukhi Numeral Recognition, Feature Extraction, Projection Histograms, Background Directional Distribution (BDD) Features, Diagonal Features, SVM classifier, RBF kernel.

1. INTRODUCTION

Optical character recognition, abbreviated as OCR, is the process of converting the images of handwritten, typewritten or printed text (usually captured by a scanner) into machine editable text or computer process able format, such as ASCII code. Applications of OCR include postal code recognition, automatic data entry into large administrative systems, banking, 3D object recognition, digital libraries, invoice and receipt processing, reading devices for blind and personal digital assistants. The three main features that characterize a good OCR system are accuracy, flexibility and speed. The basic process of an OCR system consists of phases such as: Image acquisition, preprocessing, segmentation, feature

extraction, classification and recognition and post processing as shown in figure 1.

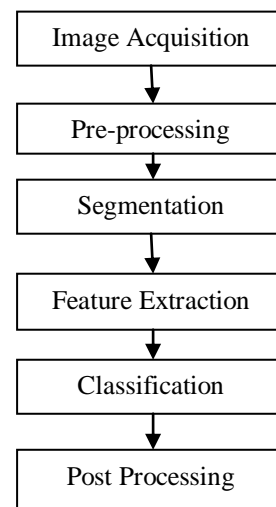


Figure 1. Basic Process of an OCR

1. Image Acquisition: In Image acquisition, the recognition system acquires a scanned image as an input image. The image should have a specific format such as JPEG, BMT etc. This image is acquired through a scanner, digital camera or any other suitable digital input device.

2. Preprocessing: The image after acquisition may carry some unwanted noise. The preprocessing stage takes in a raw image, reduces noise and distortion, removes skewness and performs skeltonizing of the image. After preprocessing phase, we get a cleaned image which is used in the segmentation phase.

3. Segmentation: The segmentation stage takes in the image after preprocessing and different logical parts, like lines of a paragraph, words of line and characters of a word are separated in this phase.

4. Feature Extraction: After segmentation, a set of features is required for each character. In feature extraction stage every character is assigned a feature vector to identify it. This vector is used to distinguish the character from other characters. Various feature extraction methods are designed like zoning, PCA, Central moments, structural features, Gabor filters and Directional Distance Distribution. Feature extraction is the process of selection of the type and the set of features. Feature extraction is the most important factor in character recognition.

5. Classification: Classification is the main decision making stage of OCR system. It uses the features extracted in the

previous stage to identify the text segment according to preset rules. Many type of classifiers are applicable to OCR like K-nearest neighbour, Neocognitions, Quadratic and SVM.

6. Post processing: The output of classification may contain some recognition errors. Post-processing methods remove these errors by making use of mostly two methods namely, dictionary lookup and statistical approach.

In the literature survey, we have found that a lot of work has been done in the recognition of isolated handwritten characters and numerals in different languages by different researchers but the work done in the field of handwritten Gurumukhi numerals is very less.

Puneet Jhaji et al. [2] have presented a feature extraction technique of zoning using K-NN and SVM for character recognition. They have obtained maximum accuracy of 73.02 using svm classifier with polynomial kernel.

Anoop Rekha [4] has presented a complete survey on different feature sets and classifiers used in offline handwritten Gurumukhi character and numeral recognition.

In [3] a box approach is proposed for extracting the features of handwritten Persian digits to achieve higher recognition accuracy and decreasing the recognition time of Persian numerals. In classification phase, support vector machine (SVM) with linear kernel has been employed as the classifier. U. Pal et al.[5] have provided a survey on all feature extraction techniques as well as training, classification and matching techniques used for recognition of machine printed and handwritten Devanagari characters and numerals state of the art from 1970s. In the literature survey [6] work done on different Indian language scripts is presented. It is found that a lot of work has been done in recognition of Devnagri and Bangla script characters, the two most popular languages in India.

G.S. Lehal et al. [7][8][10] have presented work for printed Gurmukhi script. But for Handwritten Gurmukhi Script few approaches have been practiced.

Kartar Singh Siddharth and Mahesh Jangid et al. [12] have used different feature extraction techniques for Handwritten Gurmukhi character recognition such as zoning density, Projection Histograms; distance profiles and Background Directional Distribution (BDD) features. Kartar Singh Siddharth et al. [1] has also experimented these techniques on Gurumukhi numerals and highest accuracy achieved is 99.2% when projection histogram features of different numerals are classified using svm with RBF (Radial Basis Function) kernel. Mahesh Jangid and Kartar Singh et al. [11] 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.



























In our approach we have practiced different features on different image sizes such as 30×30, 32×32 and 40×40. In diagonal features computation image is divided into equal zones each of size 4×4 pixels. Then diagonal features are extracted from pixels of each zone by moving along its diagonals. This technique is experimented on the same dataset as used in [1] by Kartar Singh Siddharth.

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 Gurumukhi numerals for our practice consists of 150 samples of each of 10 Gurumukhi numerals resulting in total 1500 samples. These samples are collected 15 different persons. Each writer contributed to write 10 samples of each of numeral of 10 different Gurumukhi 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.

Table 1. Samples of Handwritten Gurumukhi Numerals

Digit	Samples				
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					

In pre-processing, techniques like median filtration, dilation, isolated pixels removal and many other morphological operations have been applied. Before extracting the features we normalized the pre-processed numeral images to 32×32 pixel size.

3. FEATURE EXTRACTION

We have used following listed features for our experiment. Two types of features namely projection histograms and zone based diagonal features can be categorized as statistical features while third type of features (background directional distribution features) can be categorized as directional features. On the basis of these three types of features we have formed 7 feature vectors using different combinations of three basic features.

1. Projection Histograms Features

2. Bidirectional Distribution Features

3. Diagonal Features

3.1 Projection Histograms

Projection Histograms are computed by counting the number of pixels having value “1” in different directions. Projection histograms count the number of foreground pixels in specified

direction. We have used four directions of horizontal, vertical and both diagonal (left diagonal and right diagonal) traversing. Thus four types of projection histograms: horizontal, vertical, diagonal-left (left traverse) and diagonal-right (right traverse) are created in our approach. These projection histograms for a 4*4 pattern are depicted in figure 2. 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-left histogram the pixels are counted by left diagonal wise. In diagonal-right 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

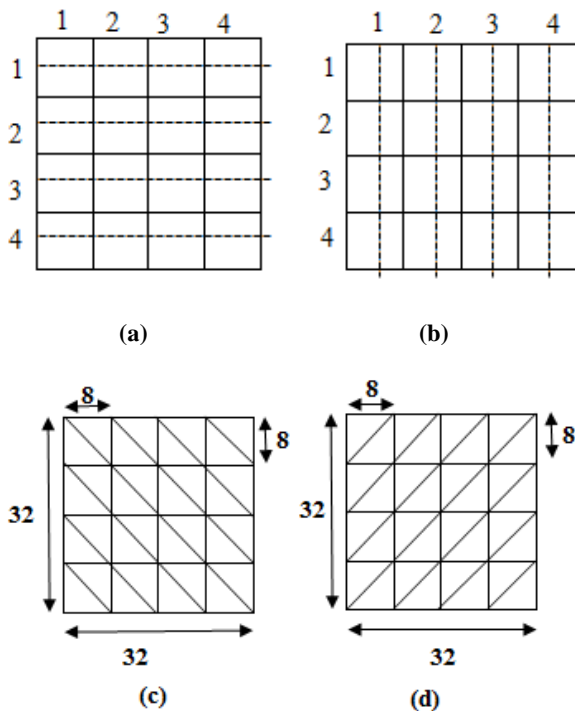


Figure 2 (a) Horizontal Histogram (b) Vertical Histogram (c) Diagonal-left Histogram (d) Diagonal-right Histogram

3.2 Background Directional Distribution (BDD) Features

For these features we have considered the directional distribution of neighboring background pixels to foreground pixels. Each image is divided into 16 equal zones each of size 8*8 pixels. For each zone 8 directional distribution features are computed. To calculate directional distribution values of background pixels for each foreground pixel masks for each directional values are used. Mask for direction 'd3' is shown in figure 3. The pixel at center 'X' is foreground pixel under consideration to calculate directional distribution values of background.

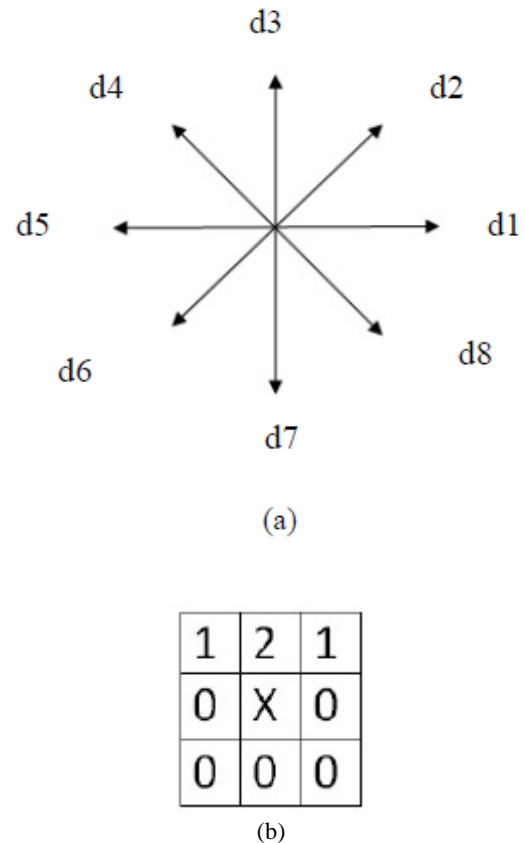


Figure 3 (a) 8-different directions for computing directional distribution (b) Mask used to compute background directional distribution in direction d3.

To compute directional distribution value for foreground pixel "X" in direction d3, for example, the corresponding mask values of neighboring background pixels will be added. Similarly we obtained all directional distribution values for each foreground pixel in 8 directions using corresponding mask. Then, all similar directional distribution values for all pixels in each zone are added. Thus finally 8 directional distribution feature values for each zone are computed. In our approach we have divided image into 16 zones. So, each numeral image is represented using 128 features

3.3 Diagonal Feature Extraction

Diagonal features are very important features in order to achieve higher recognition accuracy and reducing misclassification. These features are extracted from the pixels of each zone by moving along its diagonals as shown in Figure 4.

Algorithm for Computation of Diagonal Based Features

Step 1. Every character image of size 32*32 is divided into 64(8*8) equal zones each of size 4*4 pixels.

Step 2. The features are extracted from the pixels of each zone by moving along its diagonals.

Step 3. Each zone consists of 9 diagonals. Foreground pixels present along each diagonal are summed up in order to get a single sub feature.

Step 4. These 9 sub-features values are averaged to form a single value and placed in corresponding zone as its feature.

Step V. Corresponding to the zones whose diagonals do not have a foreground pixel, the feature value is taken as zero.

Using this algorithm, we have obtained a total of 64 features.

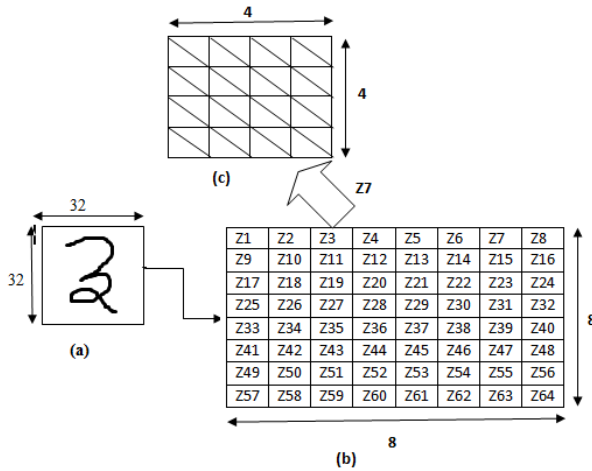


Figure 4 Diagonal Feature Extraction (a) 32×32 numeral image (b) division of image into zones (c) Diagonal direction computation in zone ‘Z7’

3.4 Feature Vectors

To form feature vectors to be used in classification we have derived 7 different feature vectors using different combinations of above described three types of basic features.

Table 2. 7 sets of feature vectors formed with different combinations of features

Feature Vector	Features	Size
FV1	Projection Histograms	190
FV2	BDD(zone size=8*8,zones=16)	128
FV3	Zone Based Diagonal Features(zone size=4*4,zones=64)	64
FV4	Histograms(FV1) + BDD(FV2)	318
FV5	Histograms(FV1) + Diagonal(FV3)	254
FV6	BDD(FV2) + Diagonal(FV3)	192
FV7	Profiles(FV1) + BDD(FV2) + Diagonal(FV3)	382

4. CLASSIFICATION

We have used SVM classifier with RBF (Radial Basis Function) kernel for recognition. The Support Vector Machine (SVM) is a new learning machine with very good generalization ability, which has been applied widely in pattern recognition and regression estimation [9]. It is primarily a two class classifier. Width of the margin between the classes is the optimization criterion, i.e. the empty area around the decision boundary defined by the distance to the nearest training pattern. These patterns called support vectors, finally define the classification function.

All the experiments are done on LIBSVM 3.0.1[15] which is multiclass SVM and select RBF (Radial Basis Function)

kernel. A feature vector set $fv(x_i)$ $i=1 \dots m$, where m is the total number of character in training set and a class set $cs(y_j)$ $j=1 \dots m, cs(y_j) \in \{0, 1, \dots, 9\}$ which defines the class of the training set, fed to Multi Class SVM.

LIBSVM implements the “one against one” approach (Kern et al., 1990) [13] for multi-class classification. Some early works of applying this strategy to SVM include, for example, Kressel (1998) [14]. If k is the number of classes, then $k(k-1)/2$ classifiers are constructed and each one trains data from two classes. For training data from the i th and j th classes, we solve the following two class classification problem:

In classification we use a voting strategy: each binary classification is considered to be a voting where votes can be cast for all data points x - in the end a point is designated to be in a class with the maximum number of votes.

In case that two classes have identical votes, though it may not be a good strategy, now we simply choose the class appearing first in the array of storing class names. LIBSVM is used with Radial Basis Function (RBF) kernel, a popular, general-purpose yet powerful kernel, denoted as

$$K(x_i, x_j) \equiv \exp(-\gamma \|x_i - x_j\|^2)$$

Now a search is applied to find the value of γ which is parameter of RBF as like find the value of c that is cost parameter of SVM using cross-validation. The value of both variance parameters are selected in the range of (0, 1] for gamma γ and (0, 1000] for cost (c) and examines the recognition rate.

4.1 5-Fold Cross Validation

In 5-fold cross validation dataset is divided into 5 equal subsets. Of these 5 equal subsets 1 subset is used for testing by classifier trained by remaining 4 subsets. Thus during every fold 1 subset is used as testing data and remaining 4 subsets as training data. Our dataset consists of 1500 samples. From these samples 300 samples are used for testing and remaining 1200 samples are used for training purpose.

The average recognition accuracy of these randomly generated 5 sets of training and testing is referred as cross validation accuracy.

5. EXPERIMENTS AND RESULTS

In our approach we have used a new technique known as diagonal features in combination with earlier approaches for handwritten Gurumukhi numeral recognition. When this recognition scheme is combined with background directional distribution (BDD) features, accuracy obtained is 99.4% which shows an improvement over earlier approaches in Literature. Second highest accuracy obtained is also 99.4 when all the features are used together. The table 2 depicts the optimized results obtained with different feature sets at refined parameters. The result variation is more sensitive to value of γ in comparison to C .

Table 3. Recognition Results with different feature sets using svm classifier

Feature	No. of Features	SVM at C=500	
		Gamma	Accuracy (%)
FV1	190	0.3-0.5	99.2
FV2	128(zone size=8×8, zones=16)	0.5	99
FV3	64(zone size=4×4, zones=64)	0.02-0.04	99.2
FV4	318[190(FV1+128(FV2))]	0.2-0.3	99.26
FV5	254[190(FV1)+64(FV3)]	0.025	99.26
FV6	192[128(FV2)+64(FV3)]	0.03-0.04	99.4
FV7	382[190(FV1)+128(FV2)+64(FV3)]	0.03-0.04	99.4

While observing the results at other values of parameter C and γ it is analysed that increasing the value of C (at small values) irrespective of any change in γ slightly increases the recognition rate, but after a certain increment normally after 64 at higher values of C the recognition rate becomes stable. After becoming stable at higher values of C, the recognition rate always changes with the slight change in γ . The optimized results are obtained at C=500 and γ value in range 2^{-5} to 2^{-1} .

6. COMPARISON WITH EARLIER APPROACHES

Table 3 shows recognition results obtained with earlier techniques applied on Gurmukhi numerals. It can be observed

Table 4. Recognition Results with earlier approaches

Proposed by	Feature Set	Classifier	Recognition Rate
Kartar Singh Siddharth et al. [1]	Zonal density and BDD(144)	SVM with RBF kernel	99.13%
Kartar Singh Siddharth et al. [1]	Projection Histograms(190)	SVM with RBF kernel	99.2%
Our Work	BDD +zone based diagonal Features(192)	SVM with RBF kernel	99.4

that a significant improvement is achieved in recognition accuracy when diagonal features are combined with background directional distribution features in comparison to zonal density features.

7. CONCLUSION AND FUTURE SCOPE

We can conclude that highest accuracy of 99.4 is obtained with FV6 and FV7 feature vector. But at the overall point of performance and efficiency, the recognition results using sixth feature set are best. It is because it has same recognition rate as FV7 but a significant reduction in number of features (382 to 192) in comparison to seventh feature set. Also it reduces time consumption and computational complexity while processing lesser number of features.

The work can be extended to increase the results by using or adding some more relevant features. More advanced classifiers as MQDF or MIL can be used and multiple classifiers can be combined to get better results.

8. REFERENCES

- [1] Kartar Singh Siddharth, Renu Dhir, Rajneesh Rani, "Handwritten Gurmukhi Numeral Recognition using Different Feature Sets," International Journal of Computer Applications (0975 – 8887) Volume 28– No.2, August 2011.
- [2] Dharamveer Sharma, Puneet Jhaji, "Recognition of Isolated Handwritten Characters in Gurmukhi Script" International Journal of Computer Applications (0975 – 8887) Volume 4– No.8, August 2010.
- [3] Omid Rashnodi, Hedieh Sajedi, Mohammad Saniee Abadeh, "Using Box Approach in Persian Handwritten Digits Recognition" International Journal of Computer Applications (0975 – 8887) Volume 32– No.3, October 2011.
- [4] Anoop Rekha, "Offline Handwritten Gurmukhi Character and Numeral Recognition using Different Feature Sets and Classifiers - A Survey" International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp. 187-191.
- [5] R. Jayadevan, Satish R. Kolhe, Pradeep M. Patil and Umapada Pal, "Offline Recognition of Devnagri Script: A Survey, IEEE Transactions On Systems, Man, And Cybernetics-Part C: Applications And Reviews, Vol.41, No. 6, November 2011,
- [6] U. Pal, B.B. Chaudhuri, "Indian Script Character Recognition: A Survey" Pattern Recognition, Elsevier, pp. 1887-1899, 2004
- [7] G.S. Lehal and Chandan Singh, "A Gurmukhi Script Recognition System" Proceedings of 15th International Conference on Pattern Recognition, Vol. 2, pp. 557-560, 2000
- [8] G.S. Lehal and Chandan Singh, "A Complete Machine printed Gurmukhi OCR System".
- [9] ZHAO Bin, LIU Yong and XIA Shao-wei, "Support Vector Machine and its Application in Handwritten Numeral Recognition" 2000 IEEE.
- [10] G.S. Lehal and Chandan Singh, "A post-processor for Gurumukhi OCR" Sadhana, Vol. 27, Part 1, pp. 99-111, 2002

- [11] 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.
- [12] Kartar Singh Siddharth, Renu Dhir, Rajneesh Rani, "Handwritten Gurmukhi Character Recognition Using Zoning Density and Background Directional Distribution Features" *International Journal of Computer Science and Information Technologies*, Vol. 2 (3) , 2011, 1036-1041
- [13] S. Knerr, L. Personnaz, and G. Dreyfus. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In J. Fogelman, editor, *Neurocomputing: Algorithms, Architectures and Applications*. Springer-Verlag, 1990.
- [14] U.H. G. Kressel. Pairwise classification and support vector machines. In B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods {Support Vector Learning}*, pages 255{268, Cambridge, MA, 1998. MIT Press
- [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>.