International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.3 (2016), pp.387-398
http://dx.doi.org/10.14257/ijmue.2016.11.3.37

Handwritten Devanagari Characters and Numeral Recognition using Multi-Region Uniform Local Binary Pattern

Prabhanjan S[1], R Dinesh[2]

[1], Research Scholar, School of Engineering and Technology, Jain University
[2], Research Supervisor, School of Engineering and Technology, Jain University,
[1]prabhan_us@rediffmail.com ,[2]dr.dineshr@gmail.com

Abstract

Automated offline handwritten character recognition of Devanagari script is a growing area of research in the field of pattern recognition. A new approach for Devanagari handwritten character / digit recognition has been proposed in this paper. This approach employs Uniform Local Binary Pattern (ULBP) operator as the feature extraction method. This operator has great performance in research areas such as texture classification and object recognition, but it has not been used in Devanagari handwritten character/digit recognition problem. The proposed method extracts both local and global features. The proposed method have two steps, in the first step image is preprocessed to remove noise and to convert it to binary image and then resizing it to a fixed size of 48x48. In the second step, ULBP operator is applied to the image to extract global features then input image is divided into 9 blocks, ULBP operator is applied to each block to extract local features. Finally, global and local features are used to train Support Vector Machine(SVM). The proposed method has been tested on large set of handwritten character and numeral database and empirical results reveals that the proposed method yields very good accuracy (98.77%) . To establish the superiority of the proposed method, it has also been compared with the contemporary algorithms. The comparative analysis shows that the proposed method out performs the existing methods.

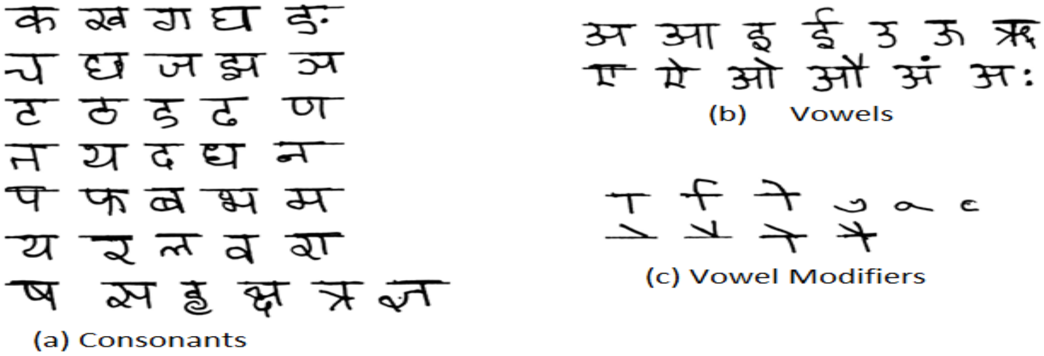
***Keywords**: Devanagari character, Handwritten character recognition, Optical Character Recognition(OCR), SVM, ULBP.*

1. Introduction

Various organization come across documents which are handwritten such as forms, checks etc.,. Such documents need to be converted and stored in digital format. Simplest form of storing them in electronic form, is to scan and store them as images. However, storing images requires lot of storage and becomes impractical as the size of the document increases. Also, it is very tedious job to manually keying in the paper document to its equivalent electronic document. Hence, Hand Printed Character Recognition (HCR)/ Optical Character Recognition (OCR) which recognizes the printed/handwritten text to recognize the characters is a natural choice. Thus Handwritten character recognition has been one of a challenging research area in the field of Computer vision and pattern recognition because of its vast application areas such as postal automation, bank cheque processing, automatic data entry etc.,. Various approaches were proposed by researchers towards handwritten recognition of Roman, Japanese, Chinese and Arabic scripts [10]. India is a multi-script, multi - lingual country, there are 22 official languages. Devanagari is the most popular script in India, spoken by more than 800 million people. Devanagari script is also used for writing Marathi, Sanskrit, Konkani and Nepali languages. Devanagari script is an alphabetic script. Devanagari script written horizontal from left to right, and characters do not have any uppercase/lowercase. It has 13 vowels and 36

ISSN: 1975-0080 IJMUE

consonants. These characters are called basic characters. In addition to basic symbols, there are set of vowel modifiers called Matra (placed to the left, right, above, or at the bottom of a character or conjunct), pure consonants (also called half letters) which when combined with other consonants yield conjuncts or compound characters. Compounding of three or four characters exits in Devanagari script. Some of the Devanagari consonants, vowels, vowel modifiers and conjuncts are shown in Figure 1. (a),(b) (c) Figure 2. respectively.



(a) Consonants

(b) Vowels

(c) Vowel Modifiers

Figure 1. Devanagari Characters (a) Consonants (b) Vowels (c) Vowel Modifiers

Figure 2. Devanagari Conjuncts

2. Literature Review

Due to its vast applications and importance of Devanagari character recognition, many work has been reported in the literature. Through literature survey, we found that many researchers have proposed methods for off-line handwritten Devanagari character recognition but many of them used only subset of Devanagari characters to evaluate their methods. Such methods, work with limited success when all characters are added into the recognition system. First research report on handwritten Devanagari characters was published in 1977 [10] on Devanagari characters and tried to solve the problem associated with them. Sharma et al. [1] proposed a method for recognition of handwritten Devanagari characters using the directional chain code information of the contour points of the characters as features. In the method[1] characters are segmented into blocks using bounding box and a CH (Chain code histogram) is computed in each of the blocks. Sharma et al. proposed a quadratic classifier-based scheme for the recognition of handwritten characters using 64-D directional chain codes and obtained 80.36 % accuracy with the 11,270 dataset size. The approach [2] used regular expressions (RE) for handwritten Devanagari character recognition. In proposed work handwritten character encoded based on chain-code features. Then, RE of stored templates is matched with encoded string. Rejected samples are then sent to a MED (minimum edit distance) classifier for recognition. They obtained 82 % accuracy on 5000 dataset. In [3] two stage classification approach for handwritten Devanagari characters were. In the first stage using structural properties like shirorekha, spine in character and second stage exploits some intersection features of characters which are fed to a feed forward neural network. To detect shirorekha and vertical bar they designed a differential distance based technique. They achieved 89.12% success rate on 50000 dataset. In [5] proposed for the recognition of handwritten Hindi Characters based on the modified exponential membership function fitted to the fuzzy sets derived from features consisting of normalized distances obtained using the Box approach. A Reuse Policy that provides guidance from the past policies used to improve the speed of the learning process and obtained 90.65 % accuracy on 4750 dataset. Pal [8] have proposed a technique for recognition of handwritten Devanagari using modified quadratic classifier using features based on directional information obtained from the arc tangent of the gradient character and obtained 94.24% accuracy on 36172 dataset. Mane [9] have presented a elastic matching with PCA based system technique towards the recognition of off-line isolated uppercase English characters and Devanagari handwritten vowels. To improve the recognition performance estimation and the utilization of Eigen deformations were used. 94.91% of recognition accuracy obtained on 3600 Devanagari vowels. Pal [10] have proposed combined approach towards the recognition of off-line Devanagari handwritten characters. Two classifiers (MQDF and SVM) based on gradient and curvature features are combined for the recognition. They experimented their approach on 36172 data set and obtained 95.13% accuracy. [11] proposed mirror image learning classifier and obtained 95.19% on gray scale curvature features on dataset of 36712 characters. Kumar[14] worked on mini batch stochastic gradient descent (SGD) based learning applied to Multilayer perceptron for recognition of isolated Devanagari handwritten character/numeral recognition using flexible zone based gradient feature extraction algorithm. They obtained 85.11% accuracy on CPAR-12 character dataset.

From the above discussion it is clear that there are few methods available for recognition of handwritten Devanagari characters/numerals. However, their recognition rates are still far from the acceptable levels. Also, most of the results reported in the existing papers were based on the very limited number of datasets and they are not evaluated on a standard datasets. There are no common datasets to evaluate methods on a common ground; thereby there was need for standard datasets. Recently, Indian Statistical Institute(ISI) have generated standard handwritten Devanagari datasets for isolated numerals and [14] have generated standard handwritten Devanagari datasets for isolated numerals and characters. In the proposed work, we have used the datasets generated by ISI [14] to evaluate the developed algorithms.

From the discussion it is also evident that most of the method proposed in literature works on subset of characters but not on the entire set hence has a limited applicability. Hence, there is a need for an effective and efficient approach for recognition of handwritten devanagri character /numeral system which produces acceptable recognition rate. In this paper, we have presented an approach based on ULBP features using SVM classifier level to improve the overall recognition accuracy of handwritten Devanagari character/numerals.

Rest of the paper is organized as follows. In section 3 we present the details of the proposed method, Experimental results were detailed in section 4, and finally concluding remarks are drawn in section 5.

3. Proposed Method

In this section, we present the proposed approach for recognition of handwritten Devanagari characters. Figure 3. demonstrates the block diagram of the proposed OCR system. At first, the input image is pre-processed. Next, features are extracted from the pre-processed character images, and finally, the character images are classified using a classifier. Details are provided in the following sub-sections.

3.1. Binarization

Generally, gray scale images will have several variations due to the nature of writing by different users, thickness of writing, smearing of ink on the paper, the tint of the paper etc. These variations pose challenges on recognition system and hence it is slightly simpler to process the bi-level image. In order to convert the gray scale image in to binary image we have employed thresholding approach. Initially, to get the optimal threshold we had employed automatic threshold using Otsu's method [15]. However, Otsu threshold did not yield good binarization, due to large variation. Hence, we have employed custom thresholding for the dataset considered. We found the threshold in the range 150 and 220 is ideal for the datasets; the range has been empirically derived. The sample binary image is shown in Figure 4.

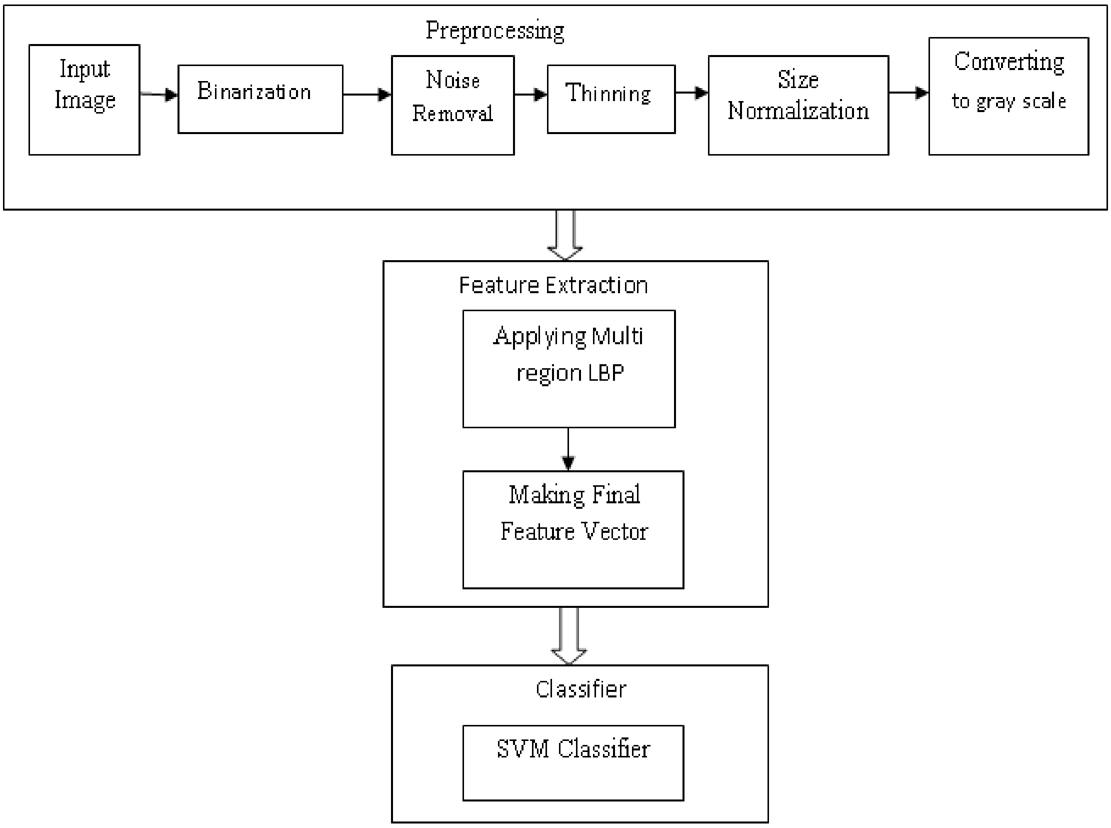


Figure 3. System Architecture

Figure 4. (a) Original Image (b) Binarized Image

3.2. Noise Removal

Most of the imaging sensors having electro mechanical limitations and hence it is impossible to eliminate the noise at the source. Hence there is a need for an effective mechanism to handle the noise at preprocessing level. The most common type of noise occurs in the image is isolated dots caused by salt and pepper kind of noise. These isolated dots need to be removed to maximize the recognition rate. In this work, we have employed various image filtering techniques to remove the nose. Images are smoothed using linear filtering operation, further to eliminate the small, isolated components and bridge the narrow opening in the characters, morphological opening and closing operations are performed.

3.3. Image Thinning

Thickness of the numeral stroke varies depending on the writer, kind of pen used and the resolution of the image while scanning. Hence making the algorithm width dependent, to make the algorithm independent of the above factors, it is essential to perform single pixel width, so that the above factors are compensated. For this purpose, in this work, we have performed the morphological thinning operation, to get single pixel width numeral. The sample output of the thinning operations is presented in Figure 5.

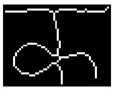


Figure 5. Thinned Image

3.4. Image Normalization

Feature extraction is very vital step in any recognition system. Most of the existing classifiers require the features to be of uniform dimension (feature vector length should be same). In order to ensure that feature vectors are of the same length, we have normalized the images through the standard image resizing operation followed by bilinear interpolation. This operation not only ensures the feature dimension is fixed; also it makes feature extraction process simple and efficient. In this work, all images have been resized (normalized) to the standard size of 48x48. The value for the resizing has been fixed to 48x48 by empirical study. The sample normalized image is shown in Figure 6.

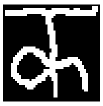


Figure 6. Normalized Image

3.5. Smoothing and Converting to Gray Scale Image

LBP method can be used only for grayscale images, therefore the input binary image should be converted to a grayscale image. Figure 7. shows a filter that used for conversion. This filter smooth the input image and convert it to a grayscale image. Figure 8. shows an example image of character (ka). After applying this filter to the input image, edges of an image will be blurred and other uniform areas remains the same. So LBP operator focus only on the edges and the non-edge parts will have no place in the extracted histogram. By extracting edges information, we can discriminate different type of characters.

0.1	**0.1**	**0.1**
0.1	**0.2**	**0.1**
0.1	**0.1**	**0.1**

Figure 7. Filter used for Smoothing and Converting Binary Image to Grayscale Image

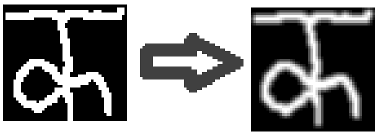


Figure 8. Applying the Filter to the Left Binary Image and Converting it to a Grayscale Image

3.6. Feature Extraction

Extracting suitable features to represent and describe the image is very vital step in any recognition system. In this research work, we have considered multi-region uniform local binary pattern features.

3.6.1. LBP Operator

[12] have proposed Local Binary Pattern (LBP) for texture description. LBP is one of the most popular and successful binary descriptors in context, object and face recognition. One of the Two important properties of LBP is robustness to monotonic gray-scale changes caused by varying illumination conditions is its computational simplicity, which makes it suitable for real-time applications. LBP operator used in variety of computer vision applications including face recognition , human detection , moving object detection and medical image analysis. Local binary pattern operator works in a 3 × 3 pixel window of an image. The pixels in this window are threshold by its center pixel value as shown in fig 1, Pixels with the values, higher than the central one (or equal to it), set with value 1 , those which are lower than the central one, set with value 0 . Thus, we get the eight-bit binary code, which describes the pixel neighborhood. Basic LBP can represent 28 =256 unique patterns.

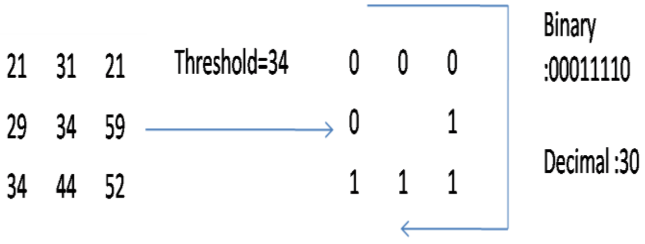
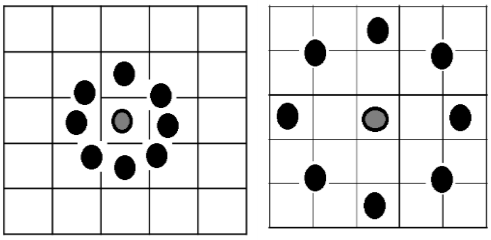


Figure 9. LBP Operator on 3x3 Window

The basic LBP operator has been extended to use circular neighborhoods because basic LBP operator is its small 3 X 3 neighborhood cannot capture dominant features with large scale structures. Hence the operator was extended to use neighborhood of different sizes. This extended LBP operator catch more structural features of entities like faces, objects, and characters [13]. In this extended LBP operator sampling points are on a circle with arbitrary radius. $_{,}$ refers to an operator with P sampling points on a circle with radius R figure 10. below shows different samples of this operator and formula 1 explains this operator. $LBP_{P,}(x,y)$ is the interpreted decimal value of pixel (x, y). g_c shows the intensity value of the central pixel and g_i plays the same role for the neighboring pixels. S(x) is calculated as follows:

$$LBP_{P,R}(x,y)=\sum_{i=0}^{P-1} s(g_i - g_c)2^i \quad (1)$$

$$S(x)= \begin{cases} 1 & \text{if}(x\geq 0) \\ 0 & \text{if}(x<0) \end{cases} \quad (2)$$



P=8,R=1 P=8,R=2

Figure 10. Extended LBP with Different P and R

multiplied by powers of two and then added to obtain a label for the center pixel. Neighborhood consists of 8 pixels, a total of 28 = 256 different labels obtained depending on the relative gray values of the center and the pixels in the neighborhood.

3.6.2. Uniform LBP

Uniform LBP is another extended version of the basic LBP. The LBP binary pattern having at most two transitions from 0 to 1 and vice versa called uniform LBP, assigned a unique LBP value. In contrast, every binary pattern that has more than two transitions is called the non uniform pattern, and it is placed in a single category. For example, the binary pattern 00110000 has two transitions it is called uniform pattern. On the other hand, the binary pattern 00110100 has four transitions, it is called non uniform pattern. Non-uniform patterns occur rarely that their probability cannot be estimated properly, thus non-uniform patterns should be discarded as a useful feature. Number of uniform patterns to the total number of patterns is 58/256 (23%) as shown empirically in [12]. Histogram of uniform LBP operator has one bin for non uniform patterns thus total number of 59 bins. Thus Uniform LBP provides a significant reduction in the feature size and also very effective in classification.

ULBP abilities in Devanagari handwritten character recognition have not been evaluated yet. Uniform LBP has been employed to obtain more compact and efficient feature vector. This operator is applied in two forms as given below.

1. The ULBP operator is applied to whole image to get a histogram with 59 bins.
2. After dividing the image into 9 equal blocks, Uniform LBP is applied to each block and the resulting histograms are concatenated.

These two obtained histograms are put along to get the final feature vector which is used in the training process. Multi-region ULBP has been used because, single region ULBP's output histogram is not discriminate enough. 590 features obtained. Using correlation based feature subset selection features(Cfs) method features are reduced to 152 features. Cfs evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. Subsets of features that are highly correlated with the class while having low inter correlation are searched using best first search.

3.7. Classification

The classification is a process assigns labels to the character images based on the features extracted and the relationships among the features. Classifier is a program that implements a specific procedure for image classification. There are two types of classification supervised and unsupervised classification. In supervised classification known samples of informational classes to classify character of unknown identity. In unsupervised classification a large number of unknown pixels divided into number of classes based on natural groupings present in image values.

In Devanagari there are similar characters. It is difficult to classify the similar characters in OCR. To overcome this difficulty many classifiers have been developed. From the literature it is found that SVM achieves high identification rate to classify characters having similar characters.

Support vector machine are computational algorithms that can be used for classification, regression or other tasks. Support Vector Machine (SVM) is primarily a classier method that performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels. Two classes are linearly separable by any hyperplane that provides no misclassification on all data points of any of the considered classes, that is, all points belonging to class A are labeled as +1, for example, and all points belonging to class B are labeled as -1. This approach is called linear classification however there are many hyper planes that might classify the same set of data as can be seen in the figure 11. below.

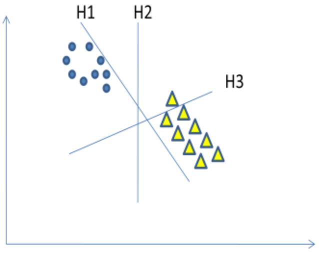


Figure 11. Hyperplanes to Classify Data

SVM is an approach to find the best separation hyperplane that provides the highest margin distance between the nearest points of the two classes (called functional margin). This approach guarantees that the larger the margin is the lower the generalization error of the classifier.

If such hyper plane exists, it provides the best separation border between the two classes and it is known as the maximum-margin hyper plane and such a linear classifier is known as the maximum margin classifier. The optimal separating hyper plane can be determined in the higher dimensional feature space by using kernel functions in the input space. Commonly used kernels

1. Linear Kernel : The Linear kernel is the simplest kernel function. It is given by the Inner Product plus optional constant ‘c’.

K(x, y) = (xT.y+c)
(3)

2. Radial Basis Function (Gaussian) Kernel :

K(x,y) = exp(-||x – y||2 /2σ2)
(4)

3. Polynomial Kernel : The Polynomial kernel is a non-stationary kernel. Polynomial kernels are well suited for problems where all the training data is normalized. For degree-d polynomials, the polynomial kernel is defined as

K(x, y) = (αxTy + c)d
(5)

Where x and y are vectors in the input space, i.e. vectors of features computed from training or test 0 is a constant trading off the≥samples, c influence of higher-order versus lower-order terms 0 , the kernel is=in the polynomial and when c called Homogeneous. SVM is used for binary classification. SVM may be extended to multiclass problems using the one-against-the-rest approach or by using the one-against-one approach.

4. Experimental Result

In this section, we present detailed experimental analysis of the proposed approach. The dataset considered for the experimentation includes both standard data sets released by ISI Kolkata, CPR12 and our own dataset generated through multiple users [7]. Overall, the number of images considered for this experimentation is 27,000 images with non-uniform distribution among images. We have developed the algorithms using Matlab and Weka.

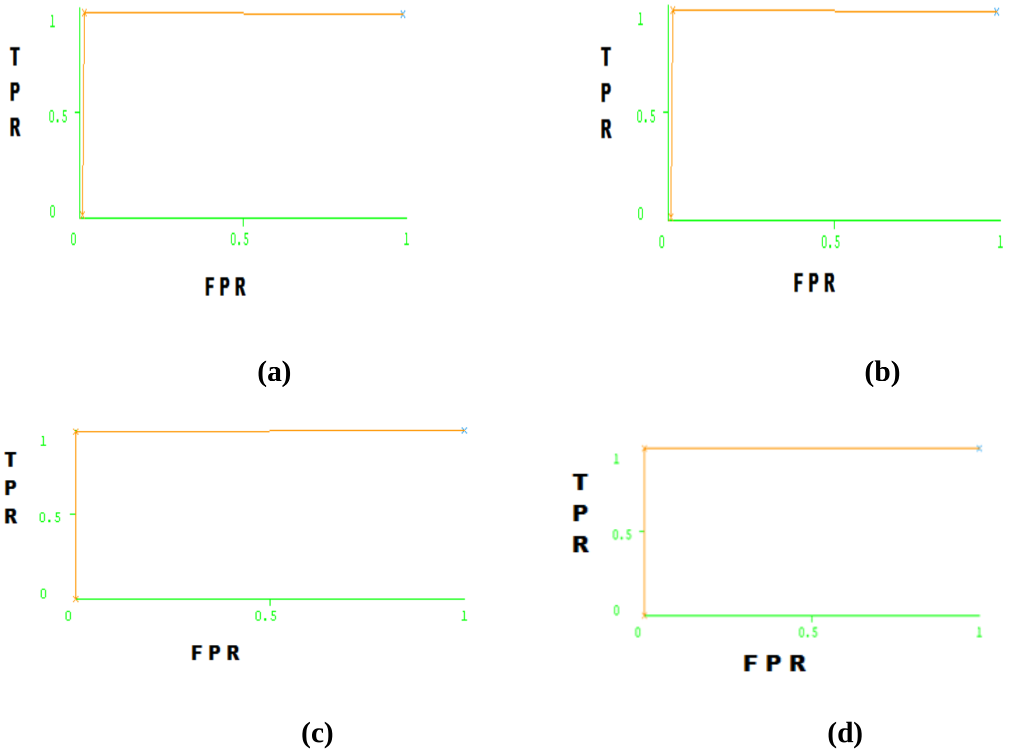
The evaluation is done using the 10 fold cross validation and the results of the proposed method show that the classification accuracy is about 98.77% using SVM classifier. This indicates that the SVM using ULBP features is well suited for Devanagari character /numeral recognition.

In order to establish the superiority of the proposed method over other contemporary algorithms, we have compared the results of the proposed method with other existing methods. Table 1. shows Comparison result. It is evident from the comparison results in Table 1. that the proposed method outperforms other existing methods in terms of computational accuracy.

Table 1. Comparison of Numerals results by researchers

Sl NO	Method proposed by	Features	Accuracy Obtained
1	Sharma[1]	Chain code Quadratic	80.36%
2	Deshpande, P.S[2]	Chain code Regular expression & MED	82%
3	Arora,S., D[3]	Structural approach	89.12 %
4	L. Malik[4]	Structural Combined	89.58%
5	Hanmandlu [5]	Vector distance Fuzzy sets	90.65%
6	Arora, S., D.[6]	Shadow & CH MLP & MED	90.74%
7	Kumar,S[7]	Gradient	94.1%
8	Pal, U.,[8]	Gradient & Gaussian filter Quadratic	94.24%
9	Mane, V.[9]	Eigen deformation Elastic matching	94.91
10	Pal, U [10]	Gradient SVM & MQDF	95.13%
11	Pal, U.,[11]	Gradient MIL	95.19%
12	Proposed Method	ULBP	98.77%

The ROC curves from the above experimentation have been given in the Figure 12. for some of the Devanagari characters like vowel A, consonants K,DHA and numeral 0. Percentage of accuracy for the a to am , ka to ha and numeral 0 to 9 as shown in Figure13. Figure 14. and Figure 15.



(a)

(b)

(c)

(d)

Figure 12. (a) Classifier ROC Curves False Positive Rate verse True Positive Rate for Devanagari Characters (a) A (b) K (c) DHA (d) Numeral 0

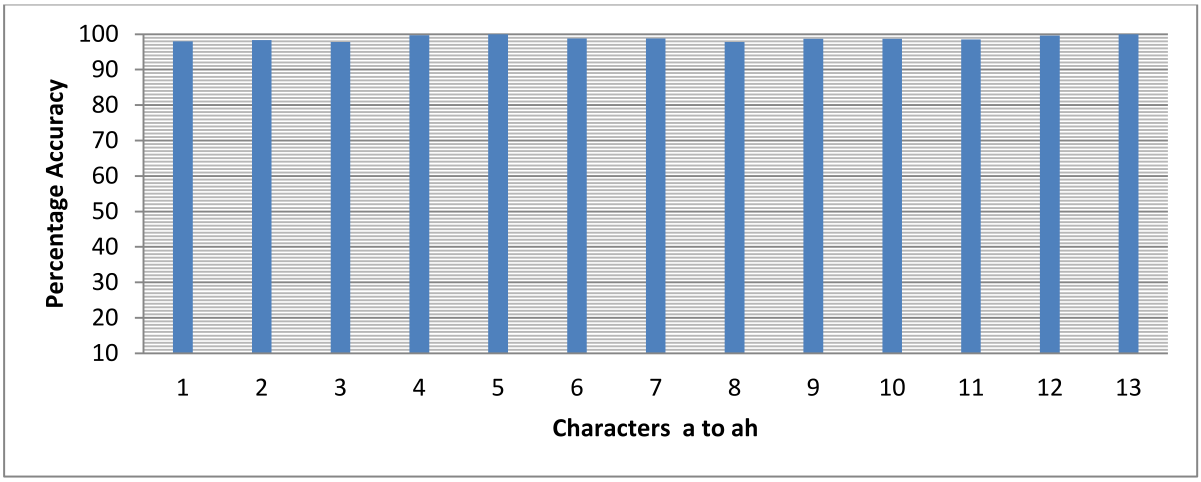


Figure 13. Classifier Accuracy for Characters a to am

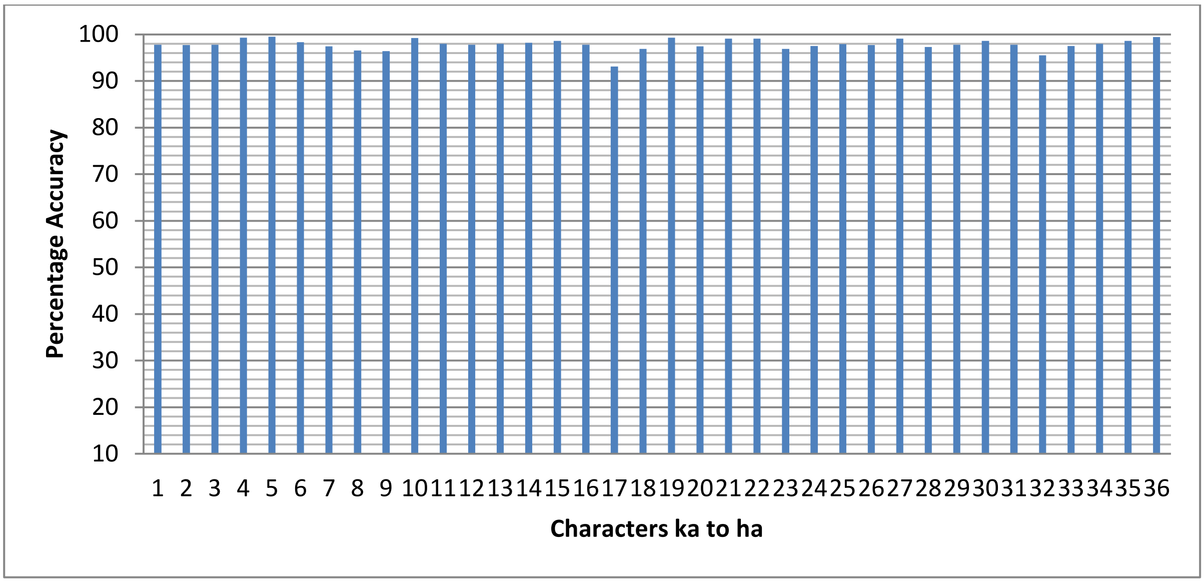


Figure 14. Classifier Accuracy for Characters ka to ha

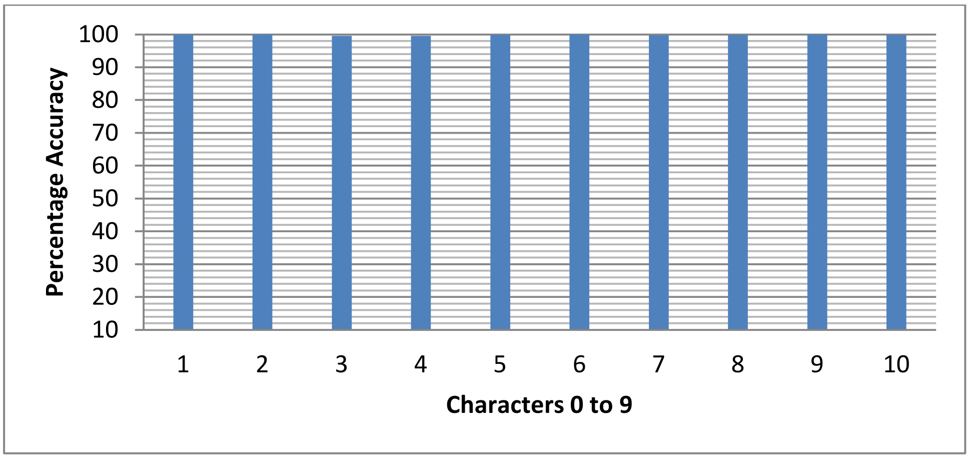


Figure 15. Classifier Accuracy for Characters 0 to 9

5. Conclusion

In this paper, a new feature extraction method for Devanagari handwritten character/numeral recognition has been proposed. Experimental results show that ULBP operator can extract very discriminating features for Devanagari handwritten character/numeral recognition. Extraordinary recognition rate (98.77%) of the proposed approach makes it one of the best methods that has been proposed in this research area.

References

[1] N. Sharma, U. Pal, F. Kimura and S. Pal, "Recognition of offline handwritten Devanagari characters using quadratic classifier", Proceedings Indian Conf. Comput. Vis. Graph, Image Process, **(2006)**.

[2] P. S.Deshpande, L. Malik and S. Arora, "Fine classification and recognition of hand written Devnagari characters with regular expressions and minimum edit distance method", Journal of Comput., vol. 3, no. 5, **(2008)**, pp. 11-17.

[3] S. Arora, D. Bhatcharjee, M. Nasipuri and L. Malik, "A two stage classification approach for handwritten Devanagari characters", Proc. Conf. Comput. Intell. Multimedia Appl., **(2007)**.

[4] S. Arora, D. Bhattacharjee, M. Nasipuri, D. K. Basu, M. Kundu and L. Malik, "Study of different

features on handwritten Devnagari character", Proc. 2nd Emerging Trends Eng. Technol., **(2009)**.

[5] M. Hanmandlu, O. V. R. Murthy and V. K. Madasu, "Fuzzy Model based recognition of handwritten Hindi characters", Proc. Int. Conf. Digital Image Comput. Tech. Appl.

[6] S. Arora, D. Bhattacharjee, M. Nasipuri, D. K. Basu and M. Kundu, "Recognition of non-compound handwritten Devanagari characters using a combination of MLP and minimum edit distance", Int. Journal of Comput. Sci. Security, vol. 4, no. 1, **(2010)**, pp. 1-14.

[7] S. Kumar, "Performance comparison of features on Devanagari handprinted dataset", Int. Journal of Recent Trends, vol. 1, no. 2, **(2009)**, pp. 33-37.

[8] U. Pal, N. Sharma, T. Wakabayashi and F. Kimura, "Off-line handwritten character recognition of Devnagari script", Proc. 9th Conf. Document Recognit.

[9] Mane, V. and L. Ragha, 2009. "Handwritten character recognition using elastic matching and PCA," Int. Conf. Adv. Comput., Commun. Control, pp: 410-415.

[10] U. Pal, S. Chanda, T. Wakabayashi and F. Kimura, "Accuracy improvement of Devnagari character recognition combining SVM and MQDF", Proceedings in Proc. 11th Int. Conf. Frontiers Handwritten. Recognit.

[11] U. Pal, T. Wakabayashi and F. Kimura, "Comparative study of Devanagari handwritten character recognition using different features and classifiers", Proc. 10th Conf. Document Anal Recognit., **(2009)**.

[12] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns", Pattern Analysis and Machine Intelligence, IEEE Transactions, vol. 24, **(2002)**, pp. 971-987.

[13] T. Ojala, M. Pietikäinen and D. Harwood, "A comparative study of texture measures with classification based on featured distributions", Pattern Recognition, vol. 29, **(1996)**, pp. 51-59.

[14] R. Kumar, K. K. Ravulakollu and Sharda University, "On the Performance of Devanagari Handwritten Character Recognition", World Applied Sciences Journal, vol. 31, no. 6, **(2014)**, pp. 1012-1019.

[15] N. Otsu, "A threshold selection method from gray-level histograms", Systems, Man and Cybernetics, IEEE Transactions, vol. 9, no. 1, **(1979)**, pp. 62-66.

Authors

S. Prabhanjan, He received BE and Masters in Computer science & Engineering from Manipal University. He is research scholar in the area of image processing at Jain University, Bangalore. His area of interest includes, compilers, image processing, Finite automata.

Dinesh Ramegowda, He comes from Mysore and currently he lives in Bangalore, Karnataka. He has completed M.Sc and Ph.D in computer science both from the University of Mysore. His area of research includes, image processing, computer vision, pattern recognition, data analytics.