Handwritten Devanagari Character Recognition Model Using Neural Network

Gaurav Jaiswal
M.Sc. (Computer Science) Department of Computer Science
Banaras Hindu University, Varanasi. India
gauravjais88@gmail.com

Abstract - **In this paper, a recognition model is described for recognizing handwritten Devanagari characters. The scanned image database of handwritten Devanagari character form several different writers was used to train and test to this classifier model. This model first preprocess (normalization, binarization, crop) then extracts the feature set. Based on the extracted feature database it classifies the characters. This model achieves the accuracy rate of recognition which range from 75% to 80%.**

***Index Terms*— Devanagari, Neural Network, Handwritten, classifier**

I. INTRODUCTION

Handwriting recognition is one of the challenging problems of Artificial Intelligence. The main challenges are large variety of handwriting style, large variety of pen type, variety of paper color where they are written. For solving this problem, several types of decision methods (including statistical method, structural method and stochastic processing) have been used along with different types of features[1][3][4][5]. Handwriting recognition for Devanagari script and its continental variation can be found in [2][11].

This paper proposed Neural Network based model to classify (recognize) handwritten Devanagari character. This system consists of three main stages – preprocessing, feature extraction and classification [fig. 1]. Input images are preprocessed before passing into the feature extraction stage. Feature vector set is extracted in the feature extraction stage using feature extraction techniques and then feed into the classifier engine. Classifier is implemented using neural network based classification algorithm.

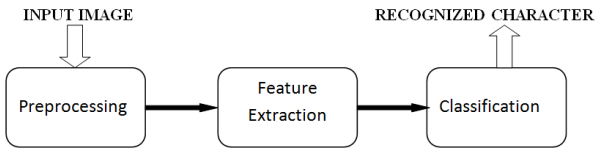

Fig. 1 : Three stage of character recognition

II. DEVANAGARI SCRIPT

Devanagari script is used to write many Indian languages such as Hindi, Marathi, Rajasthani, Sanskrit and Nepali [1][2]. The Devanagari script follows left to right fashion for writing. In Devanagari script, there are 12 vowels and 36 consonant [fig. 2]. Each Devanagari consonant has an inherent vowel (A). Vowels can be written as independent letters, or by using a variety of diacritical marks which are written above, below, before or after the consonant they belong to.

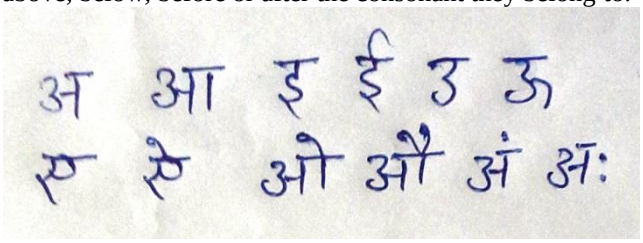

Fig. 2 : Vowels of Devanagari Script

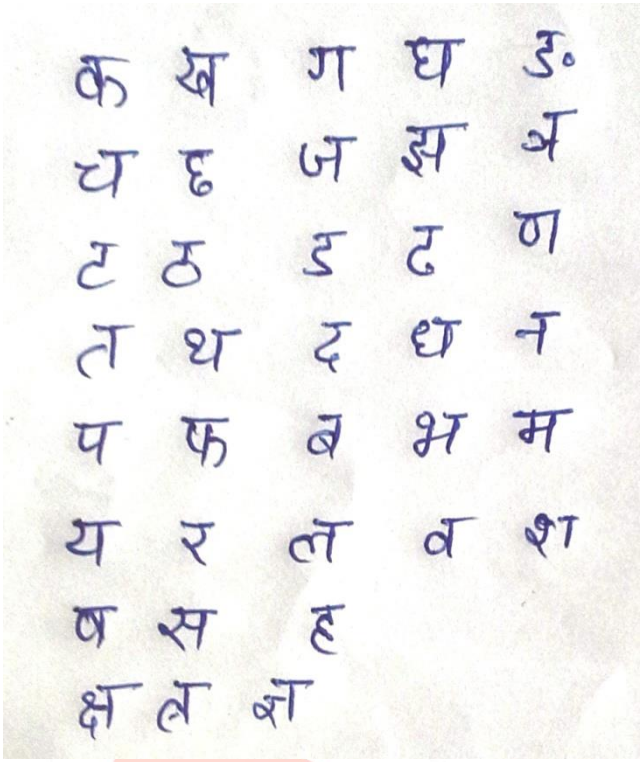

Fig. 3 : Consonant of Devanagari Script

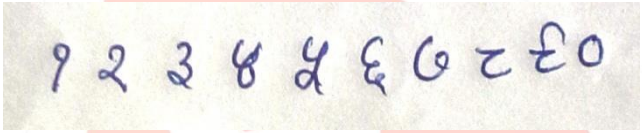

Fig. 4 : Numerals of Devanagari Script

III. IMAGE PREPROCESSING

Image preprocessing is the preliminary stage of this handwritten Devanagari character recognition model. In this stage, the following preprocessing techniques are used.

RGB to gray scale conversion

RGB image is converted into grayscale image by taking weighted summation threshold of R, G, B component of RGB image. For RGB image f(x,y), corresponding grayscale image is given by –

$$g(x,y)=0.2989*f_R+0.5870*f_G+0.1140*f_B \quad (1)$$

Where f_R, f_G, f_B are Red, Green and Blue components of the RGB image f(x; y) respectively.

Noise Removal

Noisy image pixels are removed with the help of filtering. Here, Non-linear median filtering technique is used for noise removal. For the digital image f(x; y), median filtered image is obtained as,

$$g(x; y) = \text{median}\{f(i, j) \mid (i, j) \in w\} \quad (2)$$

Where w is the neighbourhood centered around location (x,y) in the image.

Image Segmentation

Segmentation is the central problem of distinguishing object from the background. For the grayscale image f(x,y), the segmented image g(x, y) is obtained by the image binarization process as given below.

$$g(x; y) = \begin{cases} 1 \text{ if } f(x; y) >= T \\ 0 \text{ if } f(x; y) < T \end{cases} \quad (3)$$

Where, T is the threshold value and it can be obtained using the Otsu's threshold selection technique for grayscale image segmentation [10].

Image Inversion

Handwritten documents are normally written in white paper with black/blue pen. For the recognition system, we assume black pixels as a background and white pixels as the foreground. So, captured images are inverted before passing into the feature extraction stage. The inverted image of binary image f(x; y) can be obtained by the negative transformation as,

$$g(x, y) = 1 - f(x, y) \quad (4)$$

Universe of Discourse

Determining universe of discourse of the character image is finding smallest rectangle that encloses the character object. It removes extra pixels outside the convex rectangle of the character image.

Size Normalization

Size normalization is the technique of converting all the variable size input images to fixed size images. Size normalization is done so that we do not require padding of pixels at the time of feature extraction. All the input images are normalized to the predefined size of 50x70 pixels.

Image Skeletonization

Skeletonization is a process of reducing object regions in a binary image to a skeletal remainder that largely preserves the extent and connectivity of the original object. It creates single pixel wide connected object boundary that preserves Euler number of the original object.

Fig. 5 : Preprocessing step

IV. FEATURE EXTRACTION

Feature extraction stage is important stage of this model. The most important aspect of handwriting recognition scheme is the selection of good feature set, which is reasonably invariant with respect to shape variations caused by various writing styles.

There are several methods for feature extraction [7][8][9]. Here, Zone and count metric based feature extraction algorithm is used. The image (character/numeral) is divided into thirty five (10px*10 px) equal zones. Then count all white pixels (value 1) for each zone and divide by 100, which give average value of white pixels in that zone. We repeated this procedure for all the zones present in the image. There could be some zones that are empty, and then the value of that particular zone image value in the feature vector is zero. Finally 35 such features are used for feature extraction [fig. 6].

Algorithm: Zone and count metric based feature extraction system.
Input: Handwritten thinned numeral image
Output: Extracted Features for Classification and Recognition
Method Begins
Step 1: Divide the input image in to **n** equal zones.
Step 2: Compute the count of white pixel (value 1) present in the zone.
Step 3: Take average of computed value.
Step 4: Repeat this procedure sequentially for the each zone of image.
Step 7: Finally, **n** such features will be obtained for classification and recognition.
Method Ends

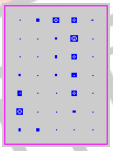

Fig. 6 : Plot of Extracted Feature set

V. CLASSIFICATION

Artificial Neural networks have emerged as an important tool for classification. The recent vast research activities in neural classification have established that neural networks are a promising alternative to various conventional classification methods [6].

Here, Multilayer feed forward neural network based classifier is used for classification. A multilayer feed forward neural network consists of a layer of input units, one or more layers of hidden units, and one layer of output units. The output from each layer is the weighted linear summation of all input vectors along with the bias term, passed through some activation function. The network weight adjustment is done by back-propagating the error of the network. The learning algorithms used for learning weights in the network are Gradient Descent with Momentum & Adaptive Learning Rate (GDMA). In this system, Input layer have 35 input node which takes extracted feature set as input. Hidden layer have 50 internal nodes and output layer have 58 output nodes (12 vowel class, 36 consonant classes, 10 numeric classes).

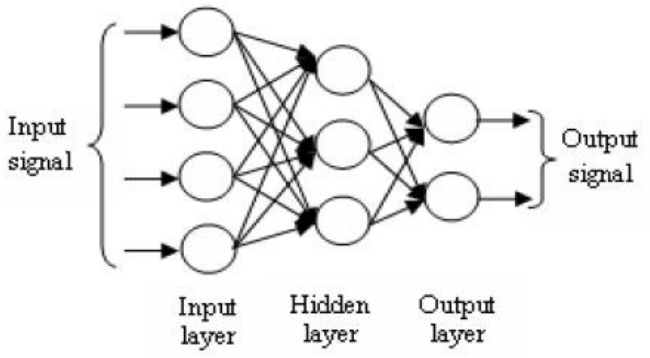

Fig. 7: Multilayer feed forward neural network

VI. EXPERIMENTAL RESULT

In order to evaluate the performance of the proposed model, we consider handwritten Devanagari characters. We have collected handwriting of 30 individual writers and total of 1740 samples are considered for devanagari characters. Here 1160 samples are used for training purpose and remaining 580 samples are used for testing. This system is simulated in MATLAB using nprtool.

Table I shows recognition results for each class of Devanagari handwritten numeral dataset.

Table II shows recognition results for each class of Devanagari handwritten vowel dataset.

Table III shows recognition results for each class of Devanagari handwritten consonant dataset.

TABLE I
RECOGNITION RESULTS FOR EACH CLASS OF NUMERALS.

Numerals	Accuracy rate(%)
१	94.4
२	83.7
३	82.4
४	83.1
५	82.5
६	82.9
७	88.1
८	97.2
९	81.4
०	97.8

TABLE III
RECOGNITION RESULTS FOR EACH CLASS OF NUMERALS.

Vowels	Accuracy rate(%)
अ	79.8
आ	93.5
इ	84.7
ई	88.5
उ	76.8
ऊ	85.3
ए	76.4
ऐ	82.8
ओ	72.4
औ	64.7

अं	66.2
अः	86.3

TABLE IIIII
RECOGNITION RESULTS FOR EACH CLASS OF NUMERALS.

Consonant	Accuracy rate(%)	Consonant	Accuracy rate(%)
क	94.3	ध	74.9
ख	80.2	न	56.2
ग	80.8	प	85.9
घ	58.2	फ	74.3
ङ	59.8	ब	73.5
च	84.3	भ	62.8
छ	65.2	म	65.7
ज	78.4	य	53.1
झ	64.6	र	89.3
ञ	57.2	ल	68.2
ट	95.8	व	74.6
ठ	82.5	श	81.3
ड	74.9	ष	72.8
ढ	87.1	स	57.3
ण	86.3	ह	86.5
त	93.7	क्ष	63.0
थ	60.2	त्र	72.7
द	65.3	ज्ञ	63.9

VI. CONCLUSION

Off-line handwriting recognition with neural network is presented and evaluated on Devanagari handwritten character datasets. Recognition system evaluated on numeral dataset gives best accuracy of 87.35% over all other datasets. In other datasets, due to high variations on writing styles, shapes and cursive nature of characters, the recognition accuracy is decreased. For vowel dataset, recognition accuracy of 79.78% is obtained and for consonant dataset, recognition accuracy of 73.47% is obtained. The total accuracy rate of this system is 77.16%.

REFERENCES

[1] Magdi A. Mohamed, and Paul Gader, "Generalized Hidden Markov Models-Part II: Application to Handwritten Word Recognition", *IEEE Trans. On Fuzzy System*, Vol. 8. No.1, February 2000.

[2] U. Pal and B. B. Chaudhari, "Indian Script Character Recogntion: a Survey", *Pattern Recogntion*, 37:1887-1899, 2004.

[3] R. Bajaj, L. Dey and S. Chaudhary, "Devanagari Numeral Recognition by Combining Decision of Multiple Connectionist Classifiers", *Sadhana*, 27:59-72, 2002.

[4] K. C. Santosh and C. Nattee, "Template-based nepali handwritten alphanumeric character recognition," *Thammasat International Journal of Science and Technology (TIJSAT), Thammasat University, Thailand*, vol. 12, no. 1, 2007.

[5] S. K. Parui and B. Shaw. Offline handwritten devanagari word recognition: An hmm based approach. *Proc. PReMI-2007(Springer)*, LNCS-4815:528–535, December 2007.

[6] Guoqiang Peter Zhang, Neural Networks for Classification: A Survey. *IEEE Transactions on systems, man, and cybernetics - part c: applications and reviews*, vol. 30, no. 4, november 2000

[7] Anil.K.Jain and Torfinn Taxt, "Feature extraction methods for character recognition-A Survey", *Pattern Recognition*, vol. 29, no. 4, pp. 641-662., 1996,

[8] K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, Jan. 2000.

[9] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, pp. 1–19, 2004.

[10] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. on Systems, Man, and cybernetics*, 9(1):62–66, March 1979.

[11] P. S. Deshpande, L. Malik, and S. Arora, "Fine classification & recognition of hand written devnagari characters with regular expressions & minimum edit distance method," *JCP*, vol. 3, no. 5, pp. 11–17, 2008.