Recognition and Editing of Devnagari Handwriting Using Neural Network

K. Y. Rajput and Sangeeta Mishra

Abstract—In this paper, a system for recognizing hand written Indian Devnagari script is presented. The system considers a handwritten images as an input, separates the lines, words and then characters step by step and then recognizes the character using artificial neural network approach, in which Creating a Character Matrix and a corresponding Suitable Network Structure is key. In addition, knowledge of how one is Deriving the Input from a Character Matrix must first be obtained before one may proceed. Afterwards, the Feed Forward Algorithm gives insight into the entire working of a neural network; followed by the Back Propagation Algorithm which compromises Training, Calculation of Error, and Modifying Weights. Once the characters are recognized they can be replaced by the standard fonts. to integrate information from diverse sources.

Index Terms—Devnagari , character matrix, recognition, neural network, network structure.

I. INTRODUCTION

Devanagari script is a logical composition of its constituent symbols in two dimensions. It has eleven vowels and thirty three simple consonants. A horizontal line is drawn on top of all characters which is referred to as the header line or *shirorekha*. A character is usually written such that it is vertically separate from its neighbors. Devanagari script has many multi-stroke characters. The data entry/ recognition mechanisms need to deal with such multi-stroke characters and also conjuncts that are made up by joining two or more characters partially.

One of the most classical applications of the Artificial Neural Network is the Character Recognition System. This

system is the base for many different types of applications in various fields, many of which we use in our daily lives. Cost

effective and less time consuming, businesses, post offices, banks, security systems, and even the field of robotics employ this system as the base of their operations. Whether you are processing a check, performing an eye/face scan at the airport entrance, or teaching a robot to pick up and object, you are employing the system of Character Recognition. One field that has developed from Character Recognition is Optical Character Recognition (OCR). OCR is used widely today in the post offices, banks, airports, airline offices, and businesses. The Address readers sort incoming and outgoing mail, check readers in banks capture images of checks for processing, airline ticket and passport readers are used for various purposes from accounting for passenger revenues to checking database records. The Form readers are used to read and process up to 5,800 forms per hour. OCR software is also used in scanners and faxes that allow the user to turn graphic images of text into editable documents. Newer applications have even expanded outside the limitations of just characters. Eye, face, and fingerprint scans used in high-security areas employ a newer kind of recognition. More and more assembly lines are becoming equipped with robots scanning the gears that pass underneath for faults, and it has been applied in the field of robotics to allow robots to detect edges, shapes, and colors[1].

Optical Character Recognition has even advanced into a newer field - Handwritten Recognition, which of course is also based on the simplicity of Character Recognition. The new idea for computers, such as Microsoft's new Tablet PC, is pen-based computing, which employs lazy recognition that runs the character recognition system silently in the background instead of in real time. Before reaching for final recognition of the character, the document is separated into line then into words and then finally into characters [2][3]. In this paper we shall be presenting a technique to recognize a devnagari hand written characters using neural network[4].

K. Y. Rajput ,Lecturer(Selection Grade), Dept of Electronics and Telecom, Thadomal Shahani Engg College, Bandra(W), Mumbai-50, India (e-mail;*rajputky9@rediffmail.com;rajputky@yahoo.co.in*)

Sangita Mishra Dept of Electronics and Telecom Thadomal Shahani Engg College, Bandra(W),Mumbai.India
(e-mail :*sngt_msr2007@yahoo.co.in*)

II. STEPS INVOLVED

The entire system can be divided in three parts.

A)   Character Separation.
B)   Preprocessing.
C)   Character recognition and editing.

III. CHARACTER SEPARATION

➢ *Line Segmentation*: Generally the problems encountered in the detection process are of two types 1) all words in a text lines are not aligned, and 2) gap between text lines is not uniform; At some places the gaps between the lines may be zero. Approach used in this system is based on the histogram at an inclination of the binary image. First we find the horizontal density histogram of a few rows in the image.

➢ *Word Segmentation*: Word boundaries are detected by looking for the vertical gaps in the segmented line, and checking them to identify the beginning and end of words.

➢ *Character Segmentation*: After detection of reference line it is removed from the word to separate out the characters again by looking for vertical gaps in the segmented word, and checking them to identify the beginning and end of character.

IV. PREPROCESSING

Before the separated characters are given to the character recognition system by neural network they need to be brought in a format that is standard and acceptable to the neural network as input. This needs preprocessing.

The aim of the paper is to develop an OCR for devanagiri script using ANN to model the characters. The Devanagiri script is used in Sanskrit, Hindi , Marathi and Nepali languages and OCR developed can be used for applications to these languages. The text is also assumed to consist of simple characters along with the headline. Various blocks of the preprocessing system are as follows:

- Image Binarisation
- Thinning of binarised image and
- Windowing

A. *Image Binarisation:*

In image binarisation , the text image which is gray scale image is converted into a binary image with each pixel taking a value of 0 or 1 depending on threshold value of the image. The technique is most commonly employed for determining the threshold involves analyzing the histogram of gray scale levels in the digitized image .

$$I(x,y) = 0 \qquad I(x,y) < t$$
$$= 1 \qquad I(x,y) >= t$$

B. *Thinning of Binarised Image:*

The characters of the text page have to be thinned prior to recognition. Thinning removes the points in such a way that only the skeleton of a branch pattern remains.

Thinning algorithms transform an object into a set of simple digital arcs. The structure is not influenced by small contour inflections. The basic approach of thinning algorithm is to delete from the object x simple border points, that have more than one neighbor in x and whose deletion does not locally disconnect x.
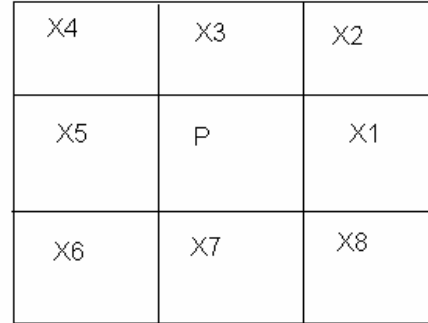
X4	X3	X2
X5	P	X1
X6	X7	X8

Figure 1. Thinning of Binarised Image.

C. *Windowing:*

Windowing the character means to bring the character to a standard image window size. This is required because after segmentation each character may have a different window size thus giving different features for the same character.

Windowing is done in two ways. First the character is fitted in a tightest possible bounding box so that the background area surroundings and which does not contain any useful information can be removed .Secondly the character size is increased to a standard size in such a way that the connectivity as well as shape of the characters are preserved. This is done, by expanding the characters to standard size, where intermediate pixels can be easily interpolated.

V. CREATING A CHARACTER RECOGNITION SYSTEM

- *C*haracter recognition by neural network

- Replacing the recognized characters by standard fonts.

• Assembling all the separated characters in the same order as they appeared in the input image to give final output.

The Character Recognition System must first be created through a few simple steps in order to prepare it for presentation into MATLAB. The matrixes of each letter of the alphabet must be created along with the network structure. In addition, one must understand how to pull the Binary Input Code from the matrix, and how to interpret the Binary Output Code, which the computer ultimately produces the Character Matrixes. A character matrix is an array of black and white pixels; the vector of 1 represented by black, and 0 by white. They are created manually by the user, in whatever size or

font imaginable; in addition, multiple fonts of the same alphabet may even be used under separate training sessions. Creating a Character Matrix First, in order to enable   a computer with the ability to recognize characters, we must first create those characters. The first thing to think about when creating a matrix is its size that will be used. Too small and all the letters may not be created, especially if you want to use two different fonts. On the other hand, if the size of the matrix is very large, it may lead to few problems: Despite the fact that the speed of computers double every third year, their may not be enough processing power currently available to run in real time. Training may take days, and results may take hours. In addition, the computer's memory may not be able to handle enough neurons in the hidden layer needed to efficient and accurately process the information. However, the number of neurons may just simply be reduced, but this in turn may greatly increase the chance of error[5].

Before the characters are recognized they must made available in a proper form, this is done by line separation, word separation and then character separation.

The steps involved in recognition of character are: Matrix generation, Thinning of Binarised Image, Windowing, Feature Extraction, Training phase and database of trained neural networks for each character.

A. Matrix generation:

A large matrix size of 30 x 30 is created, through the steps as explained above, because it may not be able to process in real time. (See Figure 2).
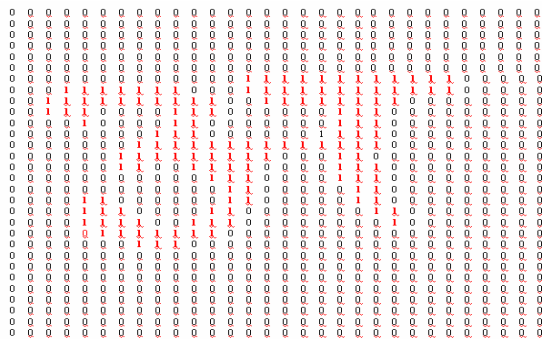

Figure 2.   Matrix size of 30 x 30 for a devnagari character

B. Neural Network:

The network receives the 900 Boolean values as a 900-element input vector. It is then required to identify the letter by responding with a 49-element output vector. The 49 elements of the output vector each of them represent a letter. To operate correctly, the network should respond with a 1 in the position of the letter being presented to the network. All other values in the output vector should be 0. In addition, the network should be able to handle the noise. In practice, the network does not receive a perfect Boolean vector as input. Specifically, the network should make least possible mistakes while classifying vectors with noise of mean 0 and standard deviation of 0.2 or less.

C. Architecture:

The neural network needs 900 inputs and 49 neurons in its output layer to identify the character. The network is a two-layer log-sigmoid/log-sigmoid network. The log-sigmoid transfer function was picked because its output range (0 to 1) is perfect for learning to output Boolean values. The hidden (first) layer has 600 neurons. This number was picked by guesswork and experience. If the network has trouble in learning, then neurons can be added to this layer. The network is trained to output 1 in the correct position of the output vector and to fill the rest of the output vector with 0's. However, noisy input vectors may not result in the network.

i) Training phase and database of trained ANNs for each character:

The neural network classification techniques such as multilayer perceptrons trained by Error Back Propagation (EBP) algorithm.
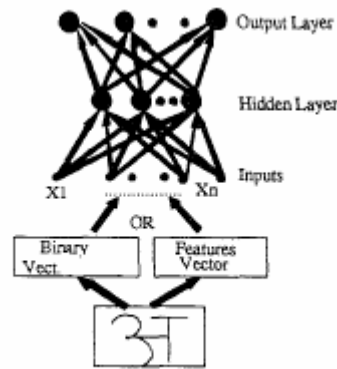

Figure3 . Multilayer perceptron network

VI. RESULTS

This section presents the results obtained at each of the above steps.

The original image taken as an input is shown in following Figure 4 .
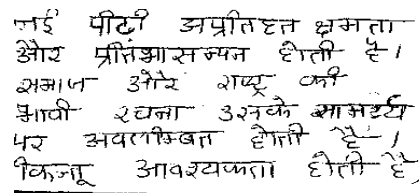

Figure 4: Sample of Image containing Devnagari hand writing.

The very first step is to separate out lines. This work is done by counting the number of black pixels per row and by plotting number of black pixels per row verses row number is as shown in the figure 5 .
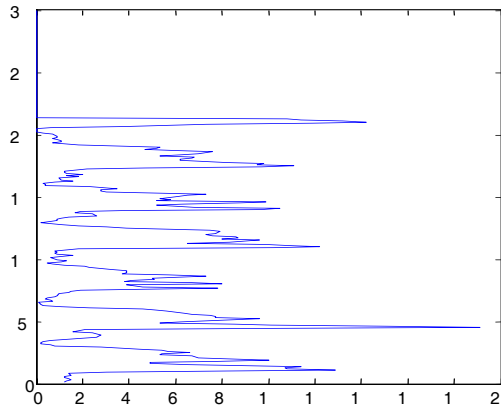
Figure 5. Histogram of image containing Devenagari handwriting.

After this step we can apply an appropriate threshold and then to separate out the lines. The result of Line separation is shown in Figure 6.
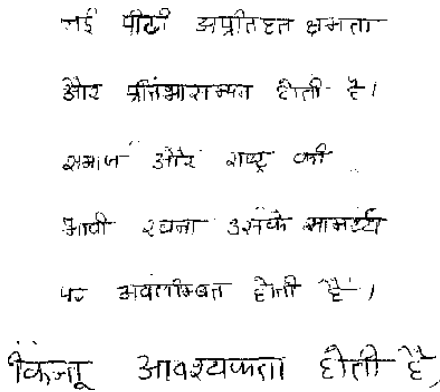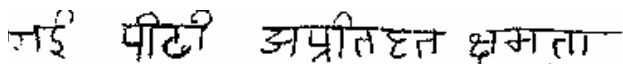

Figure 6.   Result of Line Separation.

Then word separation is carried out by vertical threshold . The result of word separation is shown in Figure 7.

Figure 7.   Result of Word Separation.

Next step is to remove shirorekha from each word. Finally a character is separated from each of the word by applying slightly higher threshold vertically. Fig. 8 shows results explaining how a word is separated in to characters.
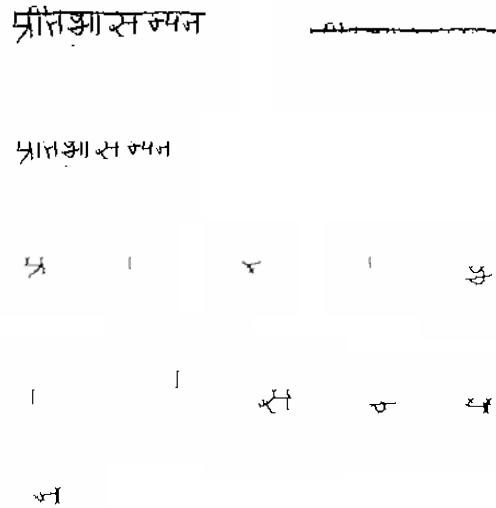

Figure 8. Complete Character Separation Results.

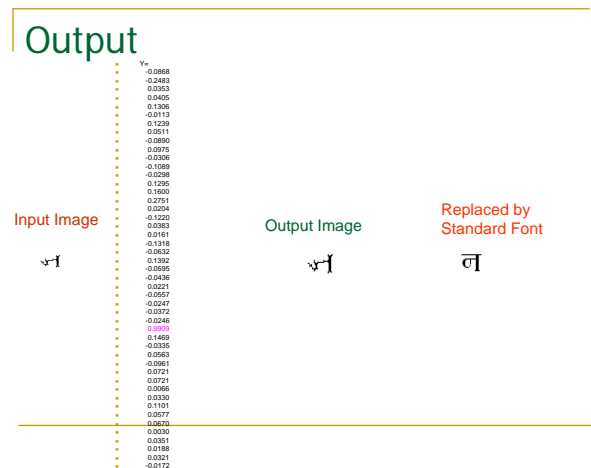Character recognition is then done using neural network.

Figure 9. Complete Process of recognition by neural network and editing.

The Figure 9 shows how each of the separated character is recognized by neural network giving an appropriate character at output. This recognized character is then replaced by a standard font. All the characters all then arranged in the same sequence as per the input image creating a final output image text.

Figure 10 shows a final output which is obtained by arranging   each of the recognized devnagari handwritten character by corresponding character in a standard font in the same order as it appeared in the input image of figure 4 .

Fig.10 Input Image of handwritten devnagari and final output obtained for the sample input of figure 4 .

VII. CONCLUSION

The method for recognition of devnagari characters using neural network presented in this paper is able to successfully recognize most of the hand writings. However, the success of the method lies in the size of database, i.e. larger the size of database used for training the neural network higher is probability of successful recognition. However the larger data base places the limit on the speed of recognition, and hence this method can be used for offline recognition.

VIII. ACKNOWLEDGMENT

We take this opportunity to express our sincere gratitude towards all those who encouraged us in carrying out the work presented in this paper particularly Prof. M.S. Agarwal, Head of the Department of Electronics and Telecommunication and Prof. H. B. Kenkre of our college. We also thank Principal and the staff of Electronics and Telecom department**,** Thadomal Shahani Engg College for all the help and support.

IX. REFERENCES

[1] Krishnamachari Jayanthi ,Akihiro Suzuki,Hiroshi Kanai,Yoshiyuki Kawazoe, Masayuki Kimura and Keniti Kido, "Devnagari character recognition using structure analysis," IEEE-1989.CH2766-4/89/0000-0363.

[2] Dileep Kumar , "An AI approach to hand written Devnagari script recognition", IIT Delhi.

[3] Yi Li,Yefeng Zheng ,and David Doermann, " Detecting text lines in handwritten documents ",The 18th International Conference on Pattern Recognition (ICPR'06).

[4] K.H. Aparna, Vidhya Subramanian, M. Kasirajan, G. Vijay Prakash, V.S. Chakravarthy, "Online Handwriting Recognition for Tamil" , Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004).

[5] Fakhraddin Mamedov and Jamal Fathi Abu Hasna, "Character recognition using neural networks" Near East University, North Cyprus, Turkey via Mersin-10, KKTC

[6] S.Hewavitharana, H.C.Fernando, "A two stage classification approach to Tamil handwriting", Tamil Internate 2002, California USA. pg.118-124.

[7] U. Bhattacharya and B. B. Chaudhuri, "Databases for Research on Recognition of Handwritten Characters of Indian Scripts ," Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition (ICDAR'05).

X. BIOGRAPHY:

K. Y. Rajput was born in Dhulia, Maharashtra on 17th June 1961. He acquired post graduation in Physics with Electronics from Institute of Science, Mumbai, India and in Electronics Design Technology, CEDT, Dr. BAM University, Aurangabad, India.

He was elected as a Senate Member in a Mumbai University and served as a member of Grievances Committee, Standing committee and Local Inquiry committees for Mumbai University, India. He was also appointed as Special Executive officer by the Government of Maharashtra, India.

His employment experience included the K. C. College, Churchgate, Mumbai, B.N. Bandodkar College, Thane and currently as Lecturer (Selection Grade) at Thadomal Shahani Engineering College, Bandra(w), Mumbai, India since 1985.

In his 24 years of teaching, he has guided over fifty undergraduate projects and over five projects at post graduate level, published six papers at national level and three papers at International level conferences.

Sangeeta Mishra was born in Mumbai, Maharashtra on 13th August 1977.She acquired post graduation in EXTC from T.S.E.C. form Mumbai.

Her employment experience included the Kirti College,Dadar ,Mumbai, T.S.E.C., Bandra, Mumbai and currently working with V.J.T.I., Matunga,Mumbai.