Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Science and Engineering

Elixir Comp. Sci. & Engg. 69 (2014) 23716-23719

ISSN: 2229-712X

# A Proposed method for designing an intelligent system for optical handwritten character recognition

Jyoti Mahajan[1,*] and Rohini Mahajan[2]

[1]Government College of Engineering & Technology, Jammu.

[2]School of Engineering, Shri Mata Vaishno Devi University, Katra

**ARTICLE INFO**

**Article history:**

Received: 7 November 2013;
Received in revised form:
20 April 2014;
Accepted: 28 April 2014;

**Keywords**

Latin-script,
Typewritten,
Recognition.

**ABSTRACT**

The accurate recognition of Latin-script, typewritten text is now considered largely a solved problem. Typical accuracy rates exceed 99%, although certain applications demanding even higher accuracy require human review for errors. Other areas—including recognition of hand printing, cursive handwriting, and printed text in other scripts (especially those with a very large number of characters)--are still the subject of active research. Recognition of cursive text is an active area of research, with recognition rates even lower than that of hand-printed text. Higher rates of recognition of general cursive script will likely not be possible without the use of contextual or grammatical information. For example, recognizing entire words from a dictionary is easier than trying to parse individual characters from script. Reading the Amount line of a cheque (which is always a written-out number) is an example where using a smaller dictionary can increase recognition rates greatly. Knowledge of the grammar of the language being scanned can also help determine if a word is likely to be a verb or a noun, for example, allowing greater accuracy. The shapes of individual cursive characters themselves simply do not contain enough information to recognize all handwritten cursive script accurately (greater than 98%). It is necessary to understand that OCR technology is a basic technology also used in advanced scanning applications. Due to this, an advanced scanning solution can be unique and patented and not easily copied despite being based on this basic OCR technology. In this paper, an intelligent system for "OPTICAL CHARACTER RECOGINITION" using Artificial Neural Network based approach and a Feature Extraction algorithm before an ANN can be applied for classification of characters which promises to provide increased efficiency for the character recognition is proposed.



## Introduction

A character can be written in a number of ways differing in shape and properties, such as Tilt, stroke, Cursively etc. Although there are different types of Fonts which have different italics and different in any commonly used Word Processing Application Software yet while perceiving any text written in a variety of ways, humans can easily recognize each character because the human perception extracts the features of the image of the character in retina that define a character's shape in an overall fashion but modeling the human perception model in machines, this task becomes a hard problem. Optical Character Recognition, usually abbreviated to OCR, is the mechanical or electronic translation of images of handwritten, typewritten or printed text into machine-editable text. The images are usually captured by a scanner. However, throughout the text, we would be referring to printed text by OCR. Data Entry through OCR has faster speed, more accuracy, and generally more efficiency than keystroke data entry. Basically, there are three types of OCR. In Offline Handwritten text is produced by a person by writing with a pencil on a paper medium and then scanned into digital format using scanner. Online Handwritten Text is written directly on a digitizing tablet using stylus. The output is a sequence of x-y coordinates that express pen position as well as other information such as pressure (exerted by the writer) and speed of writing. Machine Printed Text can be found commonly in daily use produced by offset processes, such as laser, inkjet and many more. Optical Character Recognition is used to convert different types of documents, such as scanned paper documents, PDF files or images captured by a digital camera into editable and searchable data. The OCR technology can also be used for Processing checks, Documenting library materials and Storing documents, searching text and extracting data from paper based documents

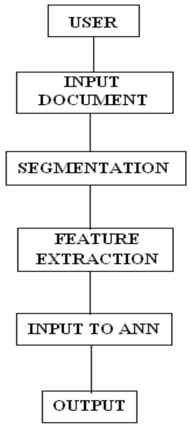
## Review Of Literature

An Optical Character recognition system based on Artificial Neural Networks (ANNs) is trained using the Back Propagation algorithm where each typed English letter is represented by binary numbers that are used as input to a simple feature extraction system whose output, in addition to the input, are fed to an ANN. After the Feed Forward Algorithm which gives workings of a neural network the Back Propagation Algorithm follows which compromises of Training, Calculating Error, and Modifying Weights. Artificial neural networks are commonly used to perform character recognition due to their high noise tolerance. The systems have the ability to yield excellent results. The feature extraction step of optical character recognition is the most important. A poorly chosen set of features will yield poor classification rates by any neural network. The most straight forward way of describing a character is by the actual raster image. Another approach is to extract certain features that still

**Tele:**
**E-mail addresses: rohinimahajan11@yahoo.in**

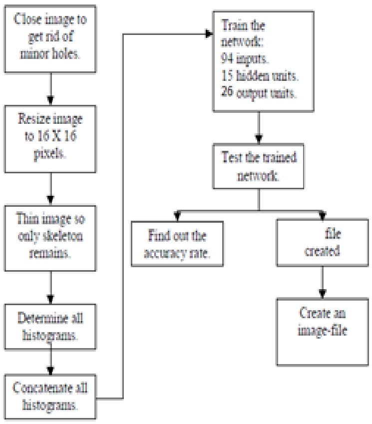
characterize the symbols, but leaves out the unimportant attributes. This may decrease the efficiency of the OCR. Despite the computational complexity involved, artificial neural networks offer several advantages in back-propagation network and classification in the sense of emulating adaptive human intelligence to a small extent. [1].Another technique focused on improving the character recognition capability of feed-forward back-propagation neural network by using one, two and three hidden layers and the modified additional momentum term. 182 English letters were collected for this work and the equivalent binary matrix form of these characters was applied to the neural network as training patterns. While the network was getting trained, then connection weights were modified at each epoch of learning. For each training sample, the error surface was examined for minima by computing the gradient descent. The experiment was started by using one hidden layer and the number of hidden layers was increased up to three and it has been observed that accuracy of the network was increased with low mean square error but at the cost of training time. The recognition accuracy was improved further when modified additional momentum term was used. [2].The architecture of the neural network is the one of a basically back propagation network with only one hidden layer (although it is the same techniques with more layers). The input layer has 35 neuron, hidden layer has 8 neurons and output layer has 26 neurons (one per letter) in this problem domain of character recognition. The training patterns are applied in some random order one by one, and the weights are adjusted using the back propagation learning law. Each application of the training set patterns is called a cycle. The patterns have to be applied for several training cycles to obtain the output error to an acceptable low value. Once the network is trained, it can be used to recall the appropriate pattern for a new input pattern. The error at the output layer itself is computed using the difference between the desired output and the actual output at each of the output units. The actual output for a given input training pattern is determined by computing the outputs of units for each hidden layer in the forward pass of the input data. The error in the output is propagated backwards only to determine the weight updates. The result is that each letter is represented as a 5 by 7 grid of Boolean values. The reliability of the neural network pattern recognition system is measured by setting the network with hundreds of input vectors with varying quantities of noise. The script file tests the network at various noise levels, and then graphs the percentage of network errors versus noise. Noise with a mean of 0 and a standard deviation from 0 to 0.5 is added to input vectors. At each noise level, 100 presentations of different noisy versions of each letter are made and the network's output is calculated. At each noise level, 100 presentations of different noisy versions of each letter are made and the network's output is calculated. The output is then passed through the competitive transfer function so that only one of the 26 outputs (representing the letters of the alphabet), has a value of 1. The number of erroneous classifications is then added and percentages are obtained. During experimentation it has been observed that Character Containing 88% Noise in letter 'A' recognized the character 'A'. The letter 'A' is almost invisible for identification. Character containing 45% noise in 'C', the letter 'C' is still very difficult to recognize but recognized correctly the letter 'C'. Character Containing 55% Noise in letter 'D', looks very similar to letter 'O', recognized correctly. Likewise all characters from 'E' to 'Z' recognized correctly. This technology is based in a field called "biometrics". The devises can be made intelligent by introducing the NN techniques can be embed the program into the hardware called "embedded system". The further scope of this can be in fingerprint identification, forensics, and signature verification/recognition of postage stamp amounts, handwritten text recognition, handwriting recognition language models, processing a handwritten sentence, document imaging or verification system, Arabic script recognition. Typewritten OCR, Handprint OCR, Cursive OCR Music OCR, Document imaging, It is check processing systems that use magnetic-ink character recognition (MICR). [3] Another technique was based on glyphs. The most basic step in OCR is to segment the input image into individual glyphs. In our approach, this is needed in two different phases, with slightly different requirements. The first is during the training stage, where segmented glyphs are presented to the human supervisor for manual classification. The other is after the network is trained and we want to recognize a new image. In this case, we need to identify each glyph in the correct sequence before extracting features from it and classifying. [4]. There are two major limitations in the current implementation which make extensive testing difficult. One is speed. The other problem is the identification of glyphs in the training stage. Currently this is done by specifying an input image, which is segmented and the user is asked to identify each glyph. The problem with this approach is that some glyphs appear far more frequently than others, which may lead to a bias in the recognition step. Another technique was based on matrices and 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. 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. The first thing to think about when creating a matrix is the size that will be used. Too small and all the letters may not be able to be created, especially if you want to use two different fonts. On the other hand, if the size of the matrix is very big, there may be a few problems that of the speed and training as results may take hours. [5] Another approach is where every character image of size 90x 60 pixels is divided into 54 equal zones, each of size 10x10 pixels. The features are extracted from each zone pixels by moving along the diagonals of its respective 10X10 pixels. Each zone has19 diagonal lines and the foreground pixels present long each diagonal line is summed to get a single sub-feature, thus 19 sub-features are obtained from the each zone. These 19 sub-features values are averaged to form a single feature value and placed in the corresponding zone. The scanned image is taken as dataset/ input and feed forward architecture is used. The structure of neural network includes an input layer with 54/69 inputs, two hidden layers each with 100 neurons and an output layer with 26 neurons. The gradient descent back propagation method with momentum and adaptive learning rate and log-sigmoid transfer functions is used for neural network training. Neural network has been trained using known dataset. [6] In the proposed research by M. Blumenstein and X. Y. Liu and B. Verma [7] two feature extraction techniques were investigated for cursive character recognition. The first is the Direction Feature (DF) technique and the second is the proposed Modified Direction Feature (MDF). The success of each feature extraction technique was tested using Multi-Layer Perceptron (MLP) and Radial Basis Function (RBF) classifiers. To summarize the character extraction process, technique first proceeded to sequentially locate all non-cursive/printed character components through the use of character component analysis. Finally, x-coordinates (vertical segmentations) for each connected character component (defined by our heuristic segmentation) were used to define the vertical boundaries of each character matrix. To locate the horizontal boundaries (top and bottom of the character matrix), the area bounded vertically (via x-coordinates or the boundaries found as a result of connected component analysis), is examined from the top and bottom. In Global Thresholding and Multiple-Pass Parsing by Joshua Goodman Harvard University [8] for instance, if in a particular cell in the chart there is some nonterminal that generates the span with high probability, and another that generates that span with low probability, then we can remove the less likely nonterminal from the cell. The less likely nonterminal will probably not be part of either the correct parse or the tree returned by the parser, so removing it will do little harm. This technique is called beam thresholding. If we use a loose beam threshold, removing only those nonterminal that are much less probable than the best nonterminal in a cell, our parser will run only slightly faster than with no thresholding, while performance measures such as precision and recall will remain virtually unchanged. On the other hand, if we use a tight threshold, removing nonterminal that are almost as probable as the best nonterminal in a cell, then we can get a considerable speedup, but at a considerable cost. The problem with beam search is that it only compares nonterminal to other nonterminal in the same cell. Thus, what we want is a thresholding technique that uses some global information for thresholding, rather than just using information in a single cell. The second kind of thresholding we consider is a novel technique, global thresholding. Global thresholding makes use of the observation that for a nonterminal to be part of the correct parse, it must be part of a sequence of reasonably probable nonterminal covering the whole sentence. Global thresholding is performed in a bottom-up chart parser immediately after each length is completed. It thus runs n times during the course of parsing a sentence of length n. We have found that, in general, global thresholding works better on simpler grammars. In some complicated grammars we explored in other work, there were systematic, strong correlations between nodes, which violated the independence approximation used in global thresholding. This prevented us from using global thresholding with these grammars. Global thresholding can be up to three times as efficient as the new beam thresholding technique, although the typical improvement is closer to 50%. When global thresholding and beam thresholding are combined, they are usually two to three times as fast as beam thresholding alone. A New Edge Detection Method based on Additions and Divisions by Sabina Priyadarshini and Gadadhar [9] uses the Sobel operator as a discrete differentiation operator that computes an approximate gradient of the image intensity function. At each point of the image, the Sobel operator is either the corresponding gradient or the norm of this vector. The Sobel method convolves an image with a small, separable and integer valued filter in horizontal and vertical direction. It produces a gradient approximation that is crude especially for high frequency variations in the image. Sobel uses 3x3 kernels, which are convolved with the original image to calculate the approximate of the derivatives. According to this method, the grey scale images are first converted into binary images. This procedure can be eliminated for a binary image. Then, the image is contracted to get the contents of the inspected region. Finally, the contents are subtracted from the inspected region to yield the boundary. The image contraction is done by dividing the image by 9. Then, the whole image is shifted in 8 directions and eight shifted images are added to the unshifted image. Next, 8 is subtracted from the grey levels of the image obtained in the previous step. If the subtraction gets a negative value, then, the intensity of that pixel is set to zero. The objective of this procedure is to obtain results whose intensity values are one at this stage. Finally, the result is multiplied by 9 to get the content having the same grey levels as that of the original binary image. The proposed algorithm is easier to implement since it involves only arithmetic and logical operations.

**Overview And Implementation**




The System is categorized into different parts. The user to the system can be anyone. The input to the system can be in the form of typed document and then it involves image acquisition that's used for removing noise from image, scaling of the image to required size, etc. Once the user gives the string in the typed or hand-written document form, then Script Separation, Line Segmentation and Character Segmentation is done. After segmentation of the given string the features are extracted. Later the steps for these are explained. The extracted feature is given as an input to the Neural Network. The output of the ANN will be the character corresponding to the given document. The system will have the capability of recognizing and updating the entered characters on-the-fly. This enables the user to rectify the errors at the time of entering the text and not when he's done entering all the characters. Pre-processing is done to remove the variability that was present in off-line handwritten characters. Hence, when performing OCR, it is common practice to convert the multilevel image into a bi-level image of black and white. Often this process, known as thresholding, is performed on the scanner to save memory space and computational effort. Deskewing is used to make the base line of the handwritten word in a horizontal direction by rotating the word in a suitable direction by a suitable angle. Some examples of techniques for correcting slope are described by Brown and Ganapathy. Slant estimation and correction is achieved by analysis of the slanted vertical projections at various angles. Scaling sometimes may be necessary to produce characters of relative size. Noise can be removed by comparing the character image by a threshold. Contour Smoothing is a technique to remove contour noise which is introduced in the form of bumps and holes due to the process of slant correction. Thinning is a process in which the skeleton of the character image is used to normalize the stroke width. Edge detection can be done using a new technique which is an improvement over the sobel's technique. Binarization is when all hand printed characters were scanned into gray scale images and each character image was traced vertically after converting the gray scale image into binary matrix. This is done in our project using global thresholding technique. Segmentation involves acquiring new document image and then either segmenting the document image into lines using the Scan Line Algorithm or segmenting each line into horizontal and vertical projections. The direction features are used to simplify each character's boundary or thinned representation through identification of individual stroke or line segments in the image. Next, in order to provide a normalized input vector to the neural network classification schemes, the new character representation was broken down into a number of windows of equal size (zoning) whereby the number, length and types of lines present in each window was determined. The line segments that would be determined in each character image are as vertical, horizontal, right diagonal and left diagonal. Aside from these four line representations, the technique also located intersection points between each type of line. To facilitate the extraction of modified direction features we find starting point and intersection point location, distinguish individual line segments, label line segment information and perform line type normalization. Classification of the alphabets is done using Neural Network. The neural net is a multi-layer perceptron with back propagation network and using log-sigmoid function. The firing rule is an important concept in neural networks and accounts for their high flexibility. A firing rule determines how one calculates whether a neuron should fire for any input pattern. It relates to all the input patterns, not only the ones on which the node was trained. A simple firing rule can be implemented by using Hamming distance technique.

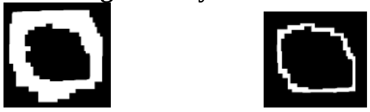
**Methodology And Results**



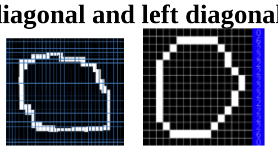

**Close the image to get rid of minor holes**

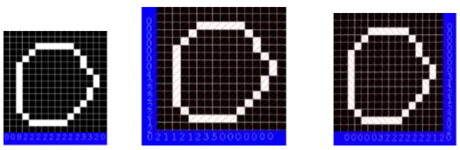
**Resize the image to 16x16 matrixes**

**Thin the image so only skeleton remains.**



**Determine all the histograms horizontal, vertical, right diagonal and left diagonal**





**Concatenation of the histograms is done and the result for an image is obtained.**

0062222222222260
0082222222223320
000000042222212021121235000000
000003222222212020221243000000

**Conclusion**

In the proposed system we have increased the DATABASE used for training the ANN, so as to enable it to recognize stylized fonts and hence demonstrated the capabilities of artificial neural network (back propagation network) implementation in recognition of characters. It is basically interfacing human with software by designing a technique simulating behavior of human neurons and classifying the patterns based on firing rules. We aim at using better algorithms for training the ANN, so as to decrease the Time complexity while handling larger databases. Better Feature Extraction techniques with both modified feature extraction technique and diagonal extraction is used.

**Future Scope**

In future this system will be used as an automated approach to character image generation, an investigation of a wider variety of global and local features and overcome the bottlenecks of previously proposed system. It can also be used to make an automated spell check of the words in a given text or document.

**References:**

[1]Sameeksha Barve., "Optical Character Recognition Using Artificial Neural Network", *Signal & Image Processing: An International Journal (SIPIJ) Vol.3, No.5 2012*

[2] Amit Choudhary and Rahul Rishi"Improving the character recognition efficiency of feedforward bp neural network", International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 1, Feb 2011DOI : 10.5121/ijcsit.2011.3107 85I

[3] Prof. S.P.Kosbatwar and Prof.S.K.Pathan "Pattern Association for character recognition by Back-Propagation algorithm using Neural Network approach" *International Journal of Computer Science & Engineering Survey (IJCSES) Vol.3, No.1, February 2012 DOI : 10.5121/ijcses.2012.3112 127*

[4] Deepayan Sarkar "Optical Character Recognition using Neural Networks" *Department of Statistics, University of Wisconsin Madison*

[5] Fakhraddin Mamedov, Jamal Fathi Abu Hasna "Character Recognition Using Neural Networks" *Near East University, North Cyprus, Turkey via Mersin-10, KKTC*

[6] J.Pradeep, E.Srinivasan and S.Himavathi "Diagonal based feature extraction for handwritten alphabets recognition system using neural network*", International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 1, Feb 2011*

[7] M. Blumenstein, X. Y. Liu and B. Verma, "A Modified Direction Feature for Cursive Character Recognition", *Int. Joint Conf. on Neural Networks (IJCNN '04), Budapest, Hungary, pp. 2983-2987, 2004*

[8] Joshua Goodman "Global Thresholding and Multiple-Pass Parsing" *Harvard University 40 Oxford St.Cambridge*

[9] Gadadhar Sahoo and Sabina Priyadarshini "A New Edge Detection Method based on Additions and Divisions" *International Journal of Computer Applications (0975 – 8887) Volume 9– No.10, November 2010*