International Journal of Emerging Science and Engineering (IJESE)
ISSN: 2319–6378, Volume-2, Issue-4, February 2014

Real Time Optical Character Recognition based on Feed Forward Networks

Dipali A. Badade, Poonam R. Deokar, Deepali B. Chavan, Manisha B. Bomble, Devidas Thosar

Abstract*— Optical Character Recognition (OCR) is the mechanical or electronic translation of images of handwritten or typewritten text (usually captured by a scanner) into machine-editable text. The main aim of this project is to design an expert system which will be best to, "Optical Character Recognition" that effectively can recognize a particular character of type format using the Feed Forward approach. OCR is a field of research in artificial intelligence, in pattern recognition and also in machine vision. Though academic research in the field that continues, the focus on OCR has been shifted to implementation of proven techniques. Optical character recognition (using optical techniques such as mirrors and lenses) and digital character recognition (using scanners and computer algorithms) were originally considered as separate fields. Because a very few applications survive that use the true optical techniques, the OCR term has been broadened now to include digital image processing as well. This system will be applicable of recognizing any number of characters including uppercase, lowercase alphabets and numerals.

Index Terms*— Optical Character Recognition, Feed Forward Networks, Image Processing, Artificial Intelligence.

I. INTRODUCTION

Optical character recognition, which usually abbreviates to OCR, is the electronic or mechanical translation of scanned images of handwritten, typewritten or printed text (usually captured by a scanner) into machine-editable text.

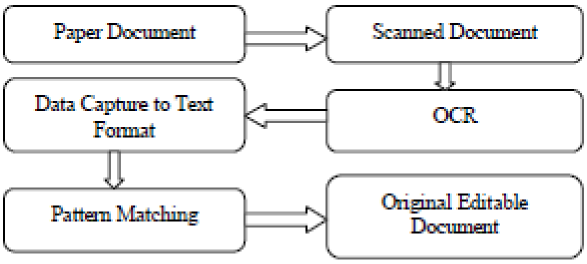

Fig. 1: Basic Concept of OCR

A. OCR Process

This section shows the basic overview of how an OCR engine processes an image to return the text contained in it:

1. The computer acquires an image of the scanned paper document.
2. The scanned document is submitted as input to the OCR engine.
3. The portions of the image are matched to shapes by OCR engine that it is instructed to recognize.
4. The OCR engine will make the given logical parameter's best guess as to which letter that a shape represents.
5. And the final OCR results are shown as text.

II. PROPOSED SYSTEM

The proposed system will be used to recognize a letter, word, sentence as well as complete paragraph. This system will be applicable of recognizing any number of characters including uppercase, lowercase alphabets and numerals.

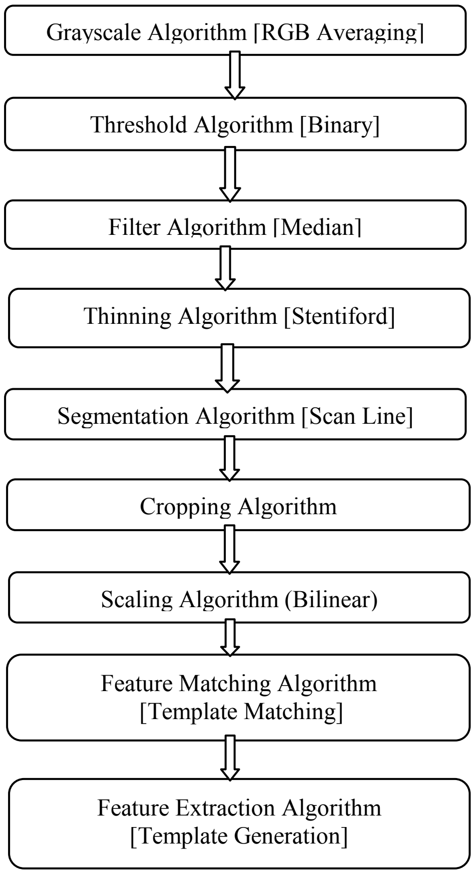

Fig. 2: Flow of Real Time OCR

This system can be used for multiple users by maintaining multiple databases and learnt characters. The central objective of this system is demonstrating the capabilities of artificial neural network implementation in recognition of characters. 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 finally integration into an off-line handwritten word recognition system. In this system we have presented a new feature extraction technique (direction feature) for the recognition of segmented handwritten characters. Handwritten recognition is used most often to describe the ability of a computer to

Manuscript received February, 2014.
Dipali Anil Badade, Computer Engineering, Pune University, Sharadchandra Pawar College of Engineering, Otur, Pune, India.
Poonam Ramesh Deokar, Computer Engineering, Pune University, Sharadchandra Pawar College of Engineering, Otur, Pune, India.
Deepali Balkrishna Chavan, Computer Engineering, Pune University, Sharadchandra Pawar College of Engineering, Otur, Pune, India.
Manisha Bhaskar Bomble, Computer Engineering, Pune University, Sharadchandra Pawar College of Engineering, Otur, Pune, India.
Prof. Devidas Thosar, Computer Engg. Department ,SPCOE, Otur, India.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

translate human writings into the text. This will be one of two ways, and it can be either by scanning of written text or by writing directly on peripheral input devices. Here we are directly writing directly on peripheral input devices. Though we are working on Real time OCR, the basic concept to design an OCR should be cleared.

III. FEED FORWARD NETWORKS

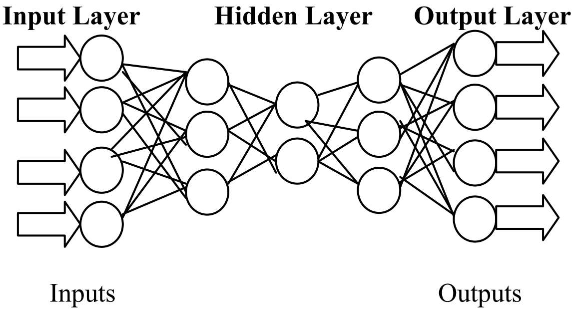

Fig. 3: Feed Forward Network

Feed-forward networks contain the following properties:

1. The perceptrons are arranged in the layers, in which the first layer taking inputs and the last layer producing outputs. The middle layers do not have any connection with the external world, and hence these are called as hidden layers.
2. Each perceptron, which is in first layer, is connected to every perceptron in the next layer. Hence information is constantly "fed forward" from one layer to the next layer, and it explains why these networks are called as feed-forward networks.
3. There is no any connection among the perceptrons which are in the same layer.

The three steps are image digitization, learning mechanism and network architecture ,which are prior to the feed forward networking.

IV. IMAGE DIGITIZATION

The document, which is put to recognition, consists of printed or handwritten characters. This document may contain pictures and colors that do not provide useful information for character recognition. The image in the document is usually processed for reducing noise and separating individual characters from document. It is convenient to show that a single character, which is noise free, has been submitted to the system for character recognition.

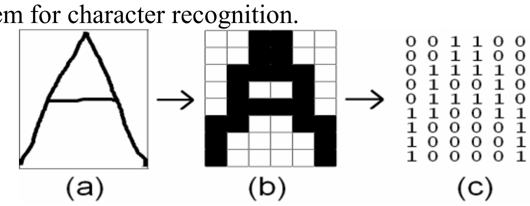

Fig. 4

The digitization process is important for the feed forward network used in this system. In this process, the input image gets sampled into a binary window which forms the input to the character recognition system. In the above figure, the alphabet A has been digitized into 6X8=48 cells which are digital, each will be having a single color, it can be either black or white. It is important to encode this information in a form that is meaningful to a computer. For this we assign a value +1 to each block pixel and 0 to each white pixel and create the binary image matrix *I* as shown in Fig. 4. This image digitization makes the input image invariant of its actual dimensions. Hence an image gets transformed into a binary matrix format of fixed pre-determined dimensions.

V. LEARNING MECHANISM

The learning mechanism of the feed forward network is described in this section. During the *training process*, the input to the neural network is the input matrix *M* defined as follows:

If I (i, j) =1Then M(i, j) =1

Else:

If I (i, j) = 0 Then M(i, j) = -1 *(1.1)*

The input matrix *M* is now fed as input to the neural network. For the *kth* character to be taught to the network, the weight matrix is denoted by *Wk*. As learning of the character progresses, it is this weight matrix that is updated. At the commencement of teaching (supervised training), this matrix is initialized to zero. Whenever a character is to be taught to the network, an input pattern representing that character is submitted to the network. The network is then *instructed* to identify this pattern as, say, the *kth* character in a *knowledge base* of characters. In accordance with this, the weight matrix *Wk* is updated in the following manner:

for all i=1 to x
{
for all j=1 to y
{
Wk(i, j) =Wk(i, j) +M(i, j)
}
} *(1.2)*

Here *x* and *y* are the dimensions of the matrix *Wk* (and *M*).

The following figure Fig. 5 which shows the digitization of three input patterns representing *S* that are presented to the system for it to learn.

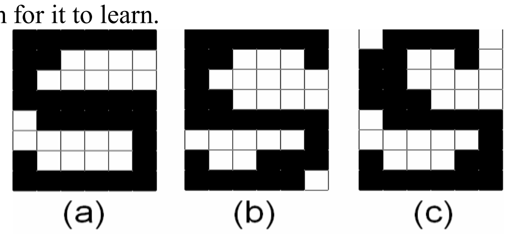

Fig. 5

Note that the patterns slightly differ from each other, just as handwriting differs from person to person (or time to time) and like printed characters differ from machine to machine.

$$Ws = \begin{array}{cccccc} 1 & 3 & 3 & 3 & 3 & 1 \\ 3 & 3 & -3 & -3 & -1 & -1 \\ 3 & -1 & -3 & -3 & -3 & -3 \\ 3 & 3 & 1 & -1 & -1 & -1 \\ -1 & 3 & 3 & 3 & 3 & 3 \\ -3 & -3 & -3 & -3 & -3 & 3 \\ 3 & -3 & -3 & -1 & 1 & 3 \\ 3 & 3 & 3 & 3 & 3 & 1 \end{array}$$

Fig. 6

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Fig. 6 gives the weight matrix, say, *WS* corresponding to the *S* alphabet. This matrix has been updated thrice to learn the alphabet *S*. It should be noted that this matrix is specific to the alphabet *S* alone. Other characters shall each have a corresponding weight matrix.
A close observation of the matrix says the following points to be noticed:

1. The matrix-elements which have higher ie. positive values are the ones which stand for the most frequently occurring image-pixels.
2. The elements which have lesser ie. negative values stand for the less frequently occurring image-pixels.

VI. NETWORK ARCHITECTURE

The overall architecture of the recognition system is shown in Fig. 7 In this system, the candidate pattern *I* is the input. The block '*M*' provides the input matrix *M* to the weight blocks *Wk* for each *k*. There are totally *n* weightblocks for the totally *n* characters to be taught (or already taught) to the system.

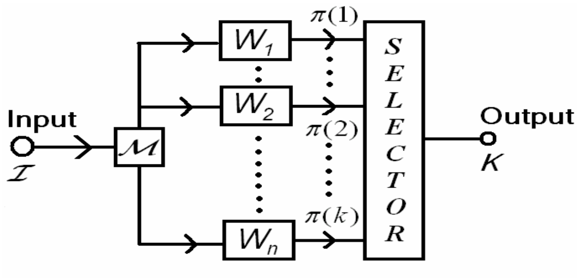

Fig. 7

The recognition of patterns is now done on basis of following statistics that will be defined next.

A. Candidate Score (Ψ)

This statistic is a product of corresponding elements of the weight matrix *Wk* of the *kth* learnt pattern and an input pattern *I* as its *candidate*. It is formulated as follows:

$$\psi(k) = \sum_{i=1}^{x} \sum_{j=1}^{y} Wk(i,j) * I(i,j) \quad (1.3)$$

It must be noted that unlike in training process where *M* was the processed input matrix, in the recognition process, the binary image matrix *I* is directly fed to the system for recognition.

B. Ideal Weight-Model Score (μ)

This statistic simply gives the sum total of all the positive elements of the weight matrix of a learnt pattern. It may be formulated as follows (with m (*k*) initialized to *0* each time).

for i=1 to x
{
for j=1 to y
{
if Wk(i, j) >0 *then*
{
$\mu(k) = \mu(k) + Wk(i, j)$
}
}
} (1.4)

C. Recognition Quotient (Q)

This statistic gives a measure of how well the recognition system identifies an input pattern as a *matching* candidate for one of its many patterns which are learnt. It is given by:

$$Q(k) = \psi(k) / \mu(k) \quad (1.5)$$

The greater the value of *Q*, the more confidence does the system bestow on the input pattern as being similar to a pattern already known to it. The classification of input patterns now follows the following trivial procedure:-

1. For an input candidate pattern *I*, calculate the recognition quotient (*Q*(*k*)) for each learnt pattern *k*.
2. Determine the value of *k* for which *Q*(*k*) has the maximum value.
3. Too low maximum value of *Q*(*k*) (say less than 0.5) indicates poor recognition. In such a case:
   - It should be concluded that the knowledge base does not contain the candidate pattern.
   OR
   - The candidate pattern should be taught to the network till it obtains a satisfactory value of *Q*(*k*).
4. Conditionally, identify the input candidate pattern as being akin to the *kth* learnt pattern OR proceed with the training for better performance.

In Fig. 7, the selector gives an output *k* by making the best selection as in Step 4 of the aforementioned algorithm. The adaptive performance of the network can easily be tested by an example: we submit two hand-drawn patterns representing *S* and *P* respectively to the system that has already learnt only the character *S*. The recognition quotient yielded by the trained system is mentioned alongside.

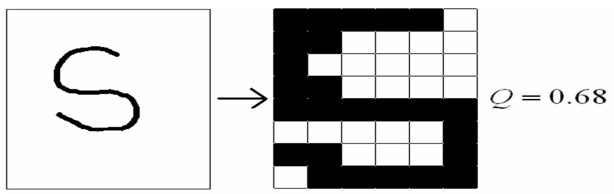

Fig. 8

Note that the pattern in Fig. 8 does not exactly appear like the three patterns of Fig. 5 that were taught to the system. However, being adaptive, the system nevertheless bestows a good quotient $Q = 0.68$ on the pattern, indicating a match. To improve recognition of this particular pattern, the same pattern can be repeatedly given as input to the system and taught to it same as before under the same label.

VII. FEATURES OF REAL TIME OCR

A. Paperless Environment

It provides the paperless environment. It allow the users to have any lengthy document electronically without any effort of typing them and listen them any time.

B. Handwritten Technique

This software can easily convert any handwritten text into the machine editable text. Since we can enter text using mouse or touch pad or touch screen in real time and the system shall immediately determine it.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

C. Multi Liangual

The software supports multiple languages (unconnected characters) like English and other languages related to English.

D. User Profile

Since the system is multi user, number of profiles can be created. This is to match the style of handwriting.

E. Real Time

The project "Real Time Optical Character Recognition" is real time. We can give the real time input using touch pad or stylus and the software immediately recognize the characters as user profile is maintained.

VIII. CONCLUSION

Optical Character Recognition is best example of Artificial Neural Networks. Feed Forward Networks are commonly used for performing recognizing due to their high noise tolerance. A simplistic approach for recognition of optical as well as handwritten characters using feed forward networks has been described.

REFERENCES

[1] "Visual Character Recognition using Artificial Neural Networks" Shashank Araokar* MGM's College of Engineering and Technology, University of Mumbai, India

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.