

# Multi-Font/Size Character Recognition and Document Scanning

Ritesh Kapoor

Bhagwan Parshuram Institute  
of Technology

Sonia Gupta

Bhagwan Parshuram Institute  
of Technology

C.M. Sharma

Bhagwan Parshuram Institute  
of Technology

## ABSTRACT

This paper presents a simple and efficient approach for the implementation of Optical Character Recognition and translation of scanned images of printed text into machine-encoded text. It makes use of four different image analysis phases followed by image detection via multi layer back propagation neural network. This paper also describes scanning the entire document and recognizing individual characters from image irrespective of their position, size and various font styles. It deals with recognition of the symbols from English language which can be easily extended for large set of symbols including UNICODE characters; this can be easily achieved by training the network for different symbol set. The phases described in this paper are simple and easy to implement.

## General Terms

Pattern Recognition, Preprocessing, Optical Character Recognizer (OCR)

## Keywords

Optical Character Recognition (OCR), ANN, Document Scanning, Image Processing

## 1. INTRODUCTION

Optical Character Recognition is the process of translating images of handwritten, typewritten, or printed text into a format understood by machines for the purpose of editing, indexing/searching, and a reduction in storage size [1]. There is a great need to accurately OCR printed materials: Much of the world information is held captive in hard-copy documents. OCR systems liberate this information by converting the text on paper into electronic form. Recognition is therefore defined as the task of text expressed in graphical format into its symbolic representation.

An emerging technique in this particular application area is the use of Artificial Neural Network (ANN) implementations with networks employing specific guides (learning rules) to update the links (weights) between their nodes. Such networks can be fed the data from the graphic analysis of the input picture and trained to output characters in one or another form [2]. ANNs have the ability to adapt, learn, generalize, cluster or organize data. There are many structures of ANNs including, Perceptron, Adaline, Madaline, Kohonen, BackPropagation and many others. Probably, BackPropagation ANN is the most commonly used, as it is very simple to implement and effective [8].

So this paper describes the use of Back Propagation neural network for the recognition of the text from the image after it undergoes various image analysis phases [3]. It also describes simple approach to scan the document and recognize individual characters from the image.

## 2. DESIGN AND IMPLEMENTATION

The recognition of individual character from the image involves image to be analyzed by four different phases namely Image crop, Image Adjustment (to constant aspect ratio), Image Downsampling (to fixed size) and Image to Matrix Transformation. However apart from these phases other phases like noise removal can also be introduced. The image to be recognized is provided as an input initially to the Image crop phase and after it is processed the output is send as an input to other phases. At the end of fourth phase the output is fed to the neural network for recognition.

All the phases described below are useful to recognize only a single symbol from the image and scanning the entire document for recognition of set of symbols is described in Document Scanning section.

Now Let us assume that the size of an image be  $S_{w,h}$  (where  $w$  represents width and  $h$  represents height for a size  $S$ ) and let the image to be a two dimensional matrix of pixel where each pixel in the image is represented as  $P_{i,j}$  (where  $i$  is  $i^{\text{th}}$  row within image and  $j$  represents  $j^{\text{th}}$  column within an image such that  $0 < i < S_h$  and  $0 < j < S_w$ ). Let us also assume a set  $Z$  such that each element  $y$  in  $Z$  is a symbol which is to be recognized.

### 2.1 Image Crop

It is initial phase of all phases and is responsible for scanning the image and determining the minimum boundaries such that it encloses the symbol completely. The region of the image within the rectangle is extracted and serves as the input for the other phase. The purpose of this phase is to extract the minimum amount of information required for recognition of the symbol. The algorithm for determining the boundaries of a rectangle is described below.

```
procedure ImageCrop(image)
  leftloop:
    for left := 1 to  $S_w - 1$  do:
      for z := 1 to  $S_h - 1$  do:
        if pixel  $P_{z,left}$  is black then
          break leftloop
        end if
```

```

    end for
  end for

rightloop:
  for right := Sw -1 to 1 do:
    for z :=1 to Sh -1 do:
      if pixel Pz,right is black then
        break rightloop
      end if
    end for
  end for

toploop:
  for top := 1 to Sh -1 do:
    for z :=1 to Sw -1 do:
      if pixel Ptop,z is black then
        break topleop
      end if
    end for
  end for

bottomloop:
  for bottom := Sh -1 to 1 do:
    for z :=1 to Sw -1 do:
      if pixel Pbottom,z is black then
        break bottomloop
      end if
    end for
  end for
end procedure

```

## 2.2 Image Adjustment

In this phase the extracted image is redrawn into minimum bounding rectangle of constant aspect ratio  $r$  aligned at the center (both horizontally and vertically). So even if the sizes of the symbols to recognize are different this phase outputs the image of the symbol with a constant ratio.

Let the size of the cropped image be  $S^l_{w,h}$  and let the constant ratio for image adjustment be  $r$  (where  $r$  is the ratio of width and height of cropped image i.e.  $r = S^l_w / S^l_h$ ); the size of the adjusted image  $S^2_{w,h}$  can be determined as follows:

```

if( $S^l_w < S^l_h \times r$ ) then
   $S^2_{w,h} = S^l_{h \times r, h}$ 
else
   $S^2_{w,h} = S^l_{w, w/r}$ 
end if

```

## 2.3 Image DownSampling

After the image is adjusted the image is downsampled or resized to constant size  $C_{w,h}$ . It is necessary to downsample the image because there is constant number of neurons in the input layer of the neural network so the input provided to the network should be constant.

The size  $C$  should neither be too small (neural network may not be trained successfully with the small sized image) and should not be too large (because neural network will require more time to get trained); so  $C$  should be of moderate size.

## 2.4 Image to Matrix Transformation

The neural network cannot be trained directly by giving image as an input. So for training the neural network the downsampled image should be transformed into matrix representation where each element in the matrix is assigned as an input to the neuron of the input layer [3].

So the image of constant size  $C_{w,h}$  is transformed into matrix  $M$  with dimensions  $w \times h$  where each entry  $M_{i,j}$  in the matrix corresponds to either 0 or 1 depending upon whether the pixel  $P_{i,j}$  is white or not in the downsampled image. So if  $P_{i,j}$  is black  $M_{i,j}$  is set to 1 else set to 0. However vice versa could also be taken.

## 2.5 Neural Network

An artificial neural network is an interconnected group of nodes, akin to the vast network of neurons in the human brain. BackPropagation ANNs contain one or more layers each of which are linked to the next layer [4]. But we will use only three layer neural network. The three layers are Input layer (all the entries in the matrix  $M$  serves as an input to this layer), Hidden Layer (number of neurons are equal to  $2/3^{\text{rd}}$  of the neurons in Input and Output layer) and Output Layer (the number of neurons is equal to the number of symbols in the set  $Z$  which are to be recognized) [7]-[11].

The network is designed and then trained over the set  $Z$  such that for every input symbol  $i$  in set  $Z$  the corresponding output neuron  $Z_i$  and has output 1 else  $Z_j=0$  where  $j \neq i$ . The network is trained till the total error is negligible. After the training is complete the network can be used to recognize the symbols in the image by performing analysis phases and then feeding the matrix to the network, the neuron  $Z_x$  with maximum value signifies the symbol  $x$  has greater probability.

Algorithm for three layer network [6]:

```

procedure InitializeNetwork
  initialize the weights in the network (often randomly)
do
  for each example  $e$  in the training set
     $o = \text{neural-net-output}(\text{network}, e)$ ; forward pass
     $t = \text{teacher output for } e$ 
    calculate error  $(t - o)$  at the output units
    compute  $\text{delta\_wh}$  for all weights from hidden layer to
      output layer; backward pass
    compute  $\text{delta\_wi}$  for all weights from input layer to
      hidden layer; backward pass continued
    update the weights in the network
  until all examples classified correctly or stopping
    criterion satisfied
  return the network
end procedure

```

After the network is initialized and trained for set of symbols  $Z$  it can be used for recognition.

All the phases for recognition of a symbol can be illustrated with the help of following example (Figure 1. Phases to recognize symbol from image.), where the output of one phase is fed as an input to another phase and finally fed to neural network for recognition.

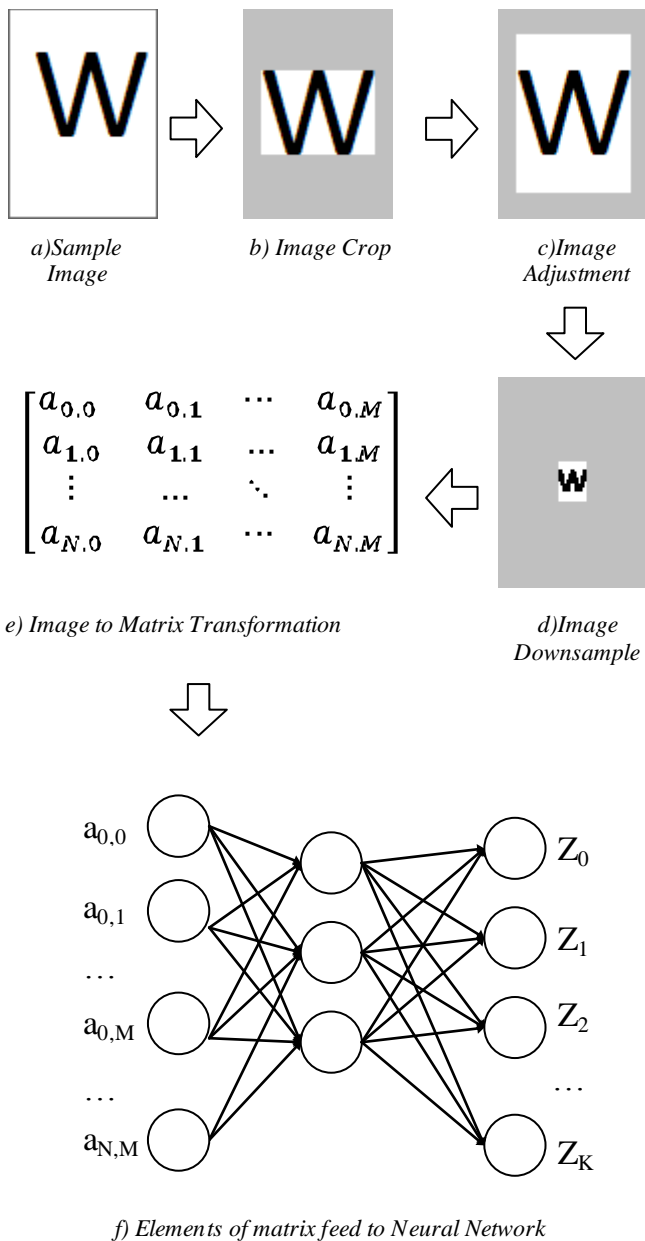


Figure 1. Phases to recognize symbol from image.

## 2.6 Document Scanning

In Document Scanning the image is scanned for the text recognition. This process is divided into two parts: Line Extraction and Symbol Extraction. In order to extract the line the document is searched for the horizontal line of pixel from the top containing no white pixel and then starting from the next line of pixels the horizontal line is searched again which does not contain any black pixel. The enclosed region is now searched for the symbols. Within this region a vertical line of pixel is searched which contains black pixel and then starting from this vertical line the next vertical line of pixel is searched

which contains no black pixel; the resulting enclosed region now contains a symbol which is send as an input to image analysis phases followed by neural network for its recognition. The entire document can be scanned for symbols (Figure 2. Scanning for symbols in document) with the help of following algorithm.

procedure Scan Document(Image)

1. Scan the image for the line which has occurrence of black pixel (Let it be top).
2. Scan the image for the line which does not have any occurrence of black pixel (Let it be bottom).
3. Scan for the leftmost line which has occurrence of black pixel (Let it be left).
4. Scan for the rightmost line which does not have any occurrence of black pixel (Let it be right).
5. Recognize the image in the rectangle [left, right, top, bottom].
6. Set left=right+1 and continue Step 3 until left less than width of image.
7. Set top=bottom+1 and continue Step 1 until top less than height of image.

end procedure

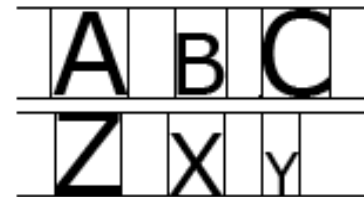


Figure 2. Scanning for symbols in document

## 3. EXPERIMENTS AND DISCUSSIONS

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

a) Set 1 (constant font size and font style)

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

b) Set 2 (variable font size and constant font style)

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

c) Set 3 (constant font size and variable font style)

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

d) Set 4 (variable font size and font style)

Figure 3. Experimental Sets

Based upon the described implementation, four different sets are taken into consideration for experimentation. All sets consist of only alphabets (i.e.  $Z = \{A, B, \dots, Z\}$ ).

The first set is rendered with a constant font size 28 and a constant font style *Tahoma*. The second set is rendered with variable font size {28, 36, 48} (applied in rotation to elements of set) and constant font style *Tahoma*. The third set is rendered with constant font size 28 and variable font styles {*Tahoma*, *Times New Roman*, *Verdana*} (applied in rotation to elements of set). And the fourth set is rendered with variable font size {28, 36, 48} and variable font styles {*Tahoma*, *Times New Roman*, *Verdana*} (applied in rotation to elements of set).

Based upon the experiments the following results were observed.

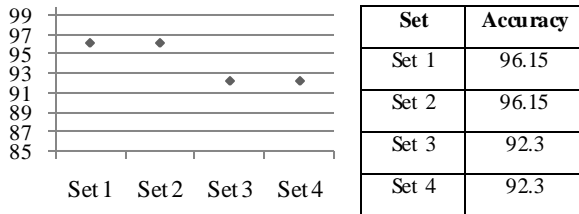


Figure 4. Results of Experiment

Table 1. Comparison Analysis

S.No.	Method proposed by	Features	Accuracy %
1.	Anita Pal et al [12]	Pre-processing and ANN	94%
2.	B.V.Dhandra et al [13]	KNN classifier	91.04%
3.	Yusuf Perwej et al [14]	MLPN	82.5%

#### 4. CONCLUSION

The implementation of the OCR described in this paper is simple and efficient. For the English alphabets and numerals the constant aspect ratio  $r$  could be set to  $5/7$  and the constant size for downsampling could be set to  $C_{15,21}$ . The document scanning approach described above can scan the document for text and convert them into machine codes. It is also capable enough to recognize large set of symbols and can successfully recognize UNICODE characters also.

#### 5. REFERENCES

[1] Optical character recognition  
[http://en.wikipedia.org/wiki/Optical\\_character\\_recognition](http://en.wikipedia.org/wiki/Optical_character_recognition)

[2] Neural Networks for Unicode Optical Character Recognition [www.projects@bangalore.com/IEEE.pdf](http://www.projects@bangalore.com/IEEE.pdf)

[3] “ $\alpha$ -Soft: An English Language OCR” (Computer Engineering and Technology (ICET), 2010 2nd International Conference on 16-18 April 2010)

[4] “Image Recognition with Neural Networks” Murat Firat <http://www.codeproject.com/KB/cs/BackPropagationNeuralNet.aspx>

[5] Backpropagation  
<http://en.wikipedia.org/wiki/Backpropagation>

[6] Creating Optical Character Recognition (OCR) applications using Neural Networks Alex Cherkasov  
[http://www.codeproject.com/KB/dotnet/simple\\_ocr.aspx](http://www.codeproject.com/KB/dotnet/simple_ocr.aspx)

[7] Neural Network Computing by R. Bharath and J. Drosen. McGraw-Hill 1994.

[8] Principles of training multi-layer neural network using Backpropagation  
[http://galaxy.agh.edu.pl/~vlsi/AI/backp\\_t\\_en/backprop.html](http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html) | Mariusz Bernacki Przemysław Włodarczyk.

[9] The backpropagation algorithm of Neural Networks – “A Systematic Introduction” by Raúl Rojas (ISBN 978-3540605058)

[10] “An Introduction To Neural Networks Pattern Learning With The Back-Propagation Algorithm” Andrew Blais, Gnosis Software, Inc.  
<http://www.ibm.com/developerworks/library/l-neural/>

[11] Principles Of Training Multi-Layer Neural Network Using Backpropagation Mariusz Bernacki Przemysław Włodarczyk  
[http://galaxy.agh.edu.pl/~vlsi/AI/backp\\_t\\_en/backprop.html](http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html) |

[12] Anita Pal, Dayashankar Singh “Handwritten English Character Recognition Using Neural Network”, International Journal of Computer Science & Communication Vol. 1, No. 2, July-December 2010

[13] B.V.Dhandra, Mallikarjun Hangarge, Gururaj Mukarambi, “Spatial Features for Handwritten Kannada and English” IJCA Special Issue on “Recent Trends in Image Processing and Pattern Recognition”

[14] Yusuf Perwej, Ashish Chaturvedi “Neural Networks for Handwritten English Alphabet Recognition” International Journal of Computer Applications (0975 – 8887)