Optical character recognition using template matching and back propagation algorithm

Swapnil Desai
CSED, Thapar University
Patiala (Punjab), India
swapnil.innovative@gmail.com

Ashima Singh
Assistant Professor, CSED, Thapar University
Patiala (Punjab), India
ashima@thapar.edu

Abstract*-- Building an effective methodology to detect characters from images with less error rate is the great task. Our aim is to furnish such an algorithm that will be able to generate error free recognition of text from the given input image which will help in document digitizing and prevention to the hand written text recognition. OCR has been in the intensive research topic for more than 4 decades, it is probably the most time consuming and labor intensive work of inputting the data through keyboard .This paper discuss about mechanical or electronic conversion of scanned images, text which contain graphics ,image captured by camera ,scanned images and the recognition of images where characters may be broken or smeared .The optical character recognition is the desktop based application developed using Java IDE and mysql as a database .We have gain 91.82% accuracy when applied on different data sets, in pre-processing we used different techniques to remove noise from the image in post processing we used dictionary for the characters which are not recognized during classification, in classification we have used the back propagation algorithm for the training of neural network, feature extraction has been performed by template matching and hamming distance. All the algorithms have been developed in java technology.*Keywords-** ***artificial intelligence, back propagation algorithm, template matching, ocr, machine learning, java api.

I. INTRODUCTION

Optical character recognition is the electronic conversion of optically processed characters. Character recognition can be offline or online, in online character recognition computer recognitions the character when it is detected.[4]

This paper discuss about the implementation of offline character recognition in java technology. The offline characters are the single characters which can be printed or handwritten.

In general the input to the OCR application is the image which are captured by camera, text images, handwritten text, typed or printed these images may have joint characters, fragmented characters, images with graphics or geometry and a noise containing text images.[1]

In this traditional method of OCR, there are three important steps which are segmentation, Feature Extraction, and Classification. In this paper we discuss how our algorithm deals with these problems. our algorithm deals with these problems.

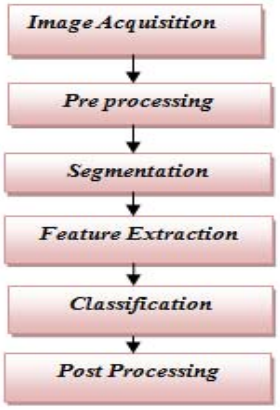

Fig. 1. Traditional steps of character recognition

In segmentation we determine the elements of an image, it is important to determine the printed data and separate it from figures and graphics, isolated characters i.e. characters which are recognized individually are segmented, this technique is easy to implement but trouble occurs when characters overlap each other. Noise containing images, graphics and geometry in the image are some other problems in the image moreover due to connected characters and merged characters in graphics, the recognition stage do not get input text for recognition [6].

The second important step is to seizure the important characteristics of each characters which distinguish each symbol, it is one of the most complex task to perform in pattern recognition, feature extraction is performed mainly through analyzing the distribution of points, through transformation, series expansion and structural analysis some feature extraction technique are template matching, transformations, distribution of points though zoning, moment, n-tuple, characteristic loci, crossings, and through structural analysis . [6]

The third important step of recognition is the classification, identifying each element characters and assigns it to the correct character class. There are two different approaches by which classification can be performed first is the decision-theoretic method, this method is used when the characters can be

numerically represented in a feature vector. Second method is structural approach this method is used when there is relationship between characteristics of characters for example if we know that the character consists of one vertical line and one horizontal line, so we can predict that the letter can be 'L' or 'T'. In this paper we have implemented the classification through neural network using back propagation algorithm.[10]

The rate of recognition of characters directly depends on the quality of image i.e. the image resolution. The scanned images are more complicated due to many possible variations in background and fonts. In this paper we will discuss the robust algorithms which work on different kind of images with different font size, colour and text.

The software is completely implemented in java technology, first image is loaded in the initial module, from where it reaches pixel extraction module in its original form. The dimensions of the image can vary in size. In pre-processing of image we applied different techniques to remove unwanted graphics, noise, and unwanted text .Image can be cropped, resized, rotated left or right, zoom in ,zoom out, and can also adjust resolution of image, if unwanted text, graphics will be removed from image extra noise in the sample image will be reduced greatly, which will increase the accuracy of recognition of text, all the module has been implemented java technology.

We used back propagation algorithm where each pixel value forms an input node and output nodes are those which are obtained from the database. The back propagation identifies the character suggest the winning neuron from the output nodes this output neuron called as the winning neuron signifies a character. Those character which are not recognized, are recognized during post processing phase where we used dictionary for the correction of words which are spelled wrong has been corrected by dictionary [10]

II. PROPOSED FRAMEWORK

This paper discuss about the implementation of optical character recognition in java technology, in the above architecture diagram. The above diagram shows the transfer of data within various modules of the project. Firstly the image gets loaded in the initial module from where it reaches the pixel extractor module in its original form i.e. in image format. The dimensions of the image becomes key data in this phase as the image processing task takes place in this form only. The pixel extractor module then brings out the equivalent array form of the image pixels. The image is converted into a grey scaled version of the input image. A corresponding array of the then grey scaled image. This array is then subjected to the segmentation module where the system evaluates each pixel of the input image and a filtering condition separates the pixels forming the text and the background. Also it classifies the text into segments and stores them as separate and disjoint array of size 120 for those pixel values that form one character.

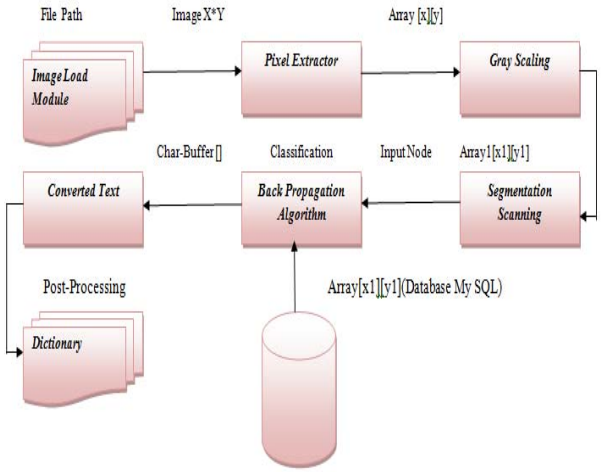

Fig. 2. Architecture Diagram

Now each of the separate arrays so formed is fed to the neural network where each pixel value forms an input node and at output nodes are those nodes which are obtained from the database. The Back propagation algorithm is used to identifies the character and suggest the winning neuron from the output nodes. This output neuron called as the winning neuron signifies a character and is sent to the text editor. In the post processing phase of character recognition we used dictionary to correct the spelling errors i.e. those characters which are not identified during recognition or classification phase

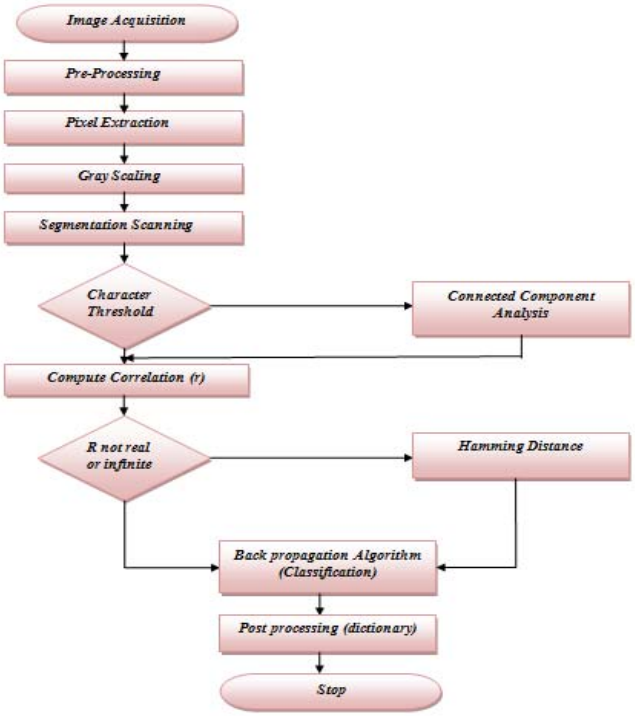

Fig. 3. Flow Chart of Proposed Framework

Character recognition process starts from image acquisition, image in acquisition is in the form of buffered image, in pre processing we perform different actions such as zoom in, zoom out, adjust resolution, rotate image left ,rotate right, crop the unwanted graphics and unwanted data from the image, after pre-processing we have performed gray scaling i.e. conversion of image in black and white image, next step is the segmentation of image vertically and horizontally for character detection, in feature extraction process we have used template matching technique after that array[x][y] is passed as the input to the back propagation algorithm, after post processing our character recognition is completed

III. PRE-PROCESSING

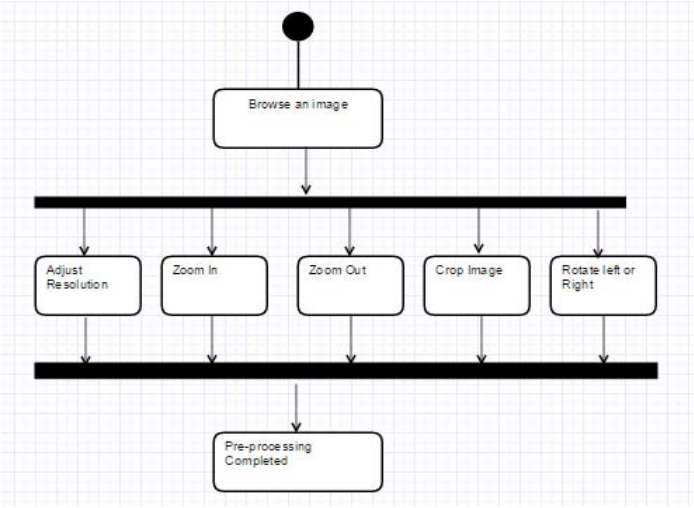

Fig. 4. Activity Diagram of Pre-Processing

We have implemented the adjust resolution feature, zoom in, zoom out, crop image, rotate left and right in java language to reduce the noise contain in the image. It has been observed that adjusting resolution also improves the accuracy of recognition of character. Cropping the image i.e. removing unwanted data, graphics also increases the accuracy to certain level.

For implementing the resolution feature in java you need to import some packages of java such as java.awt.image.buffered image, java.awt.Alphacomposite, java.awt.Graphics2D, java.awt.Image, java.awt.image.BufferedImage, java.io.File, java.io.IOException, javax.swing.ImageIcon.

```
Pseudo code
1 Take integer type image value in RGB.

int imageTy= preserveAlpha ?
BufferedImage.TYPE_INT_RGB :
BufferedImage.TYPE_INT_ARGB
2 Take Image as Buffered Image. Pass width and height in
the BufferedImage function.

  BufferedImage BI = new BufferedImage(scaledWidth,
scaledHeight, imageTy)
3 Using Buffered Image create Graphics2D object
  Graphics2D gr = BI.createGraphics()
  if (preserveAlpha) {
                gr.setComposite(AlphaComposite.Src);
        }
        gr.drawImage(Image,  0,  0,  scaledWidth,
scaledHeight, null)
        gr.dispose()
4 Return the Scaled image.
```

IV. GRAY SCALING

Image captured by camera often contains of colored images for the image processing it is required that the image should be in gray scaled, It consists of three colors namely red (r),green(g),blue(b).grey scale image only consists intensity information, this type of image are also known as black and white, it is composed exclusively of shades of grey, with black at weakest intensity and white with the strongest, gray scale images measures the intensity of light in the single band of electromagnetic spectrum. The intensity of pixels has been represented as a range from 0 as black and 1 as white, -having fractional value in between. In java image will be accepted as buffered image instead of file

```
Algorithm
Begin
        Method (Buffered Image image)
            1. Calculate the width and height of image.
            2. For all i=1 to i<height
                 For all j=1 to j<width
                 f(x, y)
                                  = 0.3 ∗ r(x, y) + 0.601
                                  ∗ g(x, y) + 0.117
                                  ∗ b(x, y)
                  end for
                 end for
end
```

after applying this function we get gray scale image, the transformation is applied for all the pixels it is represented as matrix of gray level intensities $Ihxj = [f(x, y)]$ .hxj where h and j denote the number of rows i.e. the height of the image and the number of the columns i.e. The width of the image respectively. $f(x,y) \in GL = \{0,1, \ldots, L-1\}$,the set of all gray levels, where L is the total number of gray levels in the image. Such a grey level act as an input to the recognition.

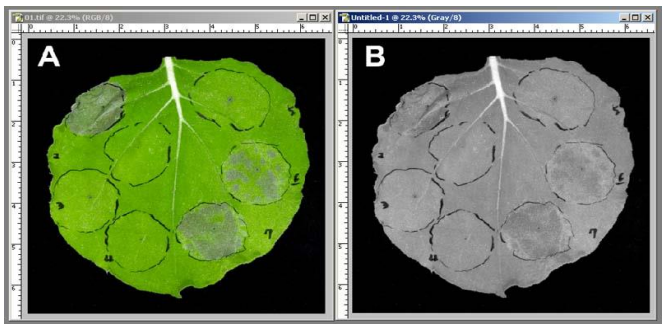

Fig. 5. Gray Scaling Conversion of Image

V. SEGMENTATION

In segmentation we determine the elements of an image, it is important to determine the printed data and separate it from figures and graphics, isolated characters i.e. characters which are recognized individually are segmented, this process is easy to implement but due to connected characters and merged character in graphics problem occurs. Pre-processed image is further divided into lines, words and then characters. First the image is segmented line wise to detect line, after detection offline word segmentation is performed; when we get the word we perform character segmentation

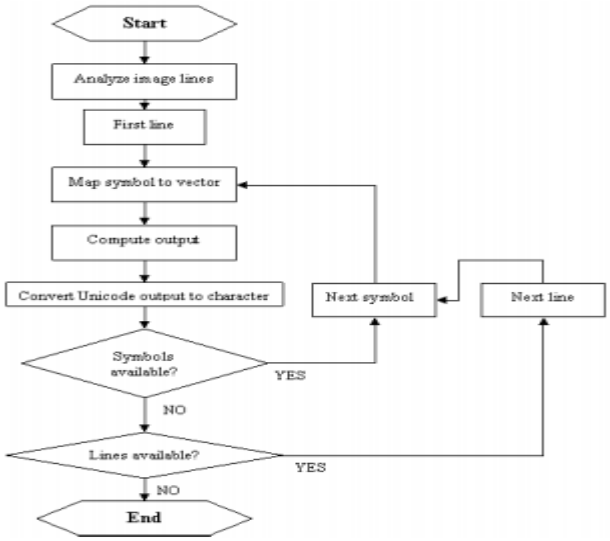

Fig. 6. Flow Diagram of Segmentation

VI. FEATURE EXTRACTION

Template matching is the simple and common feature extraction method, it is also known as correlation, in these individual characters pixels matrix are used for feature detection. A correlation function R, depends on the linear cross – correlation, it is the pattern matching procedure, this works on the test data set and the template stored in the database. The character with highest correlation value is selected as the best match for that character.

$$R = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \quad (1)$$

In the above formula A is the character extracted and B is the character which is stored in the database. $\bar{A}$ and $\bar{B}$ are the average of A and B. A problem can occur if the denominator is close to 0, in this case the correlation may fail or gain undesirable matching, to solve this problem we use hamming distance.

Hamming distance find divergence or dissimilarity between two binary codes, vectors or matrices. Hamming distance can be performed as (XOR) between two matrices, the template with least hamming distance has lesser divergence and is selected to be the best match .

$$HD = \frac{sum(A \oplus B)}{N} \quad (2)$$

VII. CLASSIFICATION

Back propagation algorithm

Back propagation is supervised learning algorithm. Back propagation is also known as "back propagation of errors". It reduces the error by propagating it backward, it is based on the error correction. It works in two pass.

Pass 1:

1. In the forward propagation training input data is applied to the nodes of the network. To generate the propagation at the output nodes.
2. Finally we get outputs produced by the response of the network .During forward propagation weight of the network is fixed.

Pass 2:

1. Synaptic weights are adjusted with respect to the error correction rule in the backward propagation.
2. From the desired response we subtract actual response to produce an error.
3. The error calculated is propagated backward in the network to adjust the synaptic weights, the process continues till we get the desired output.

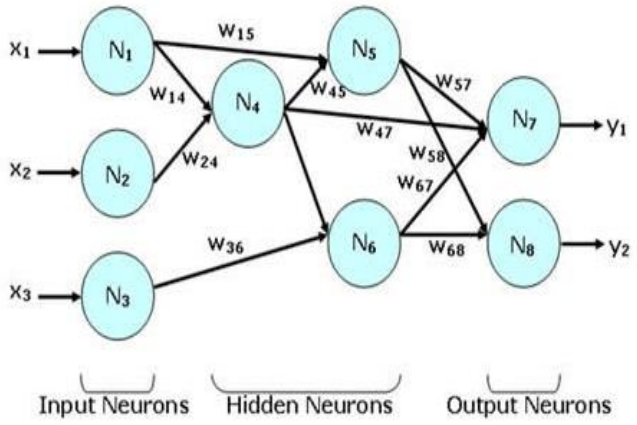

Fig. 7. Layers of Back Propagation

VIII. POST-PROCESSING THROUGH DICTIONARY

After the classification stage, when characters are grouped in words and words from sentences, the un-identified characters are mistakenly interpreted as some other characters. In optical text

recognition application, even the best application will not detect 100% correct identification of all characters.

for example 'T' can be recognized as 'L' or 'P' can be recognized as 'F' because during classification array matrix of bit pattern are almost similar so when the input node is compared with output node in neural network the winning neuron can be classified wrong, to overcome such problem we have implemented the dictionary in post processing phase of recognition.

Table I Error Correction

Error in words	After Correction
MOLHER	MOTHER
PINAL	FINAL
PINAL	FINAL

You can use Rapid Spell (JAR) File for implementing dictionary in java.

IX. EXPREMENTAL RESULTS

When correlation and hamming distance is applied in feature extraction as template matching technique with the functioning of back propagation algorithm as classification we get the optimum results for optical character recognition, it has been tried out on several types of images by varying the image size, namely the image of different pixels such as 32x32, 64x64.It has also been tested on low pixels images 4x4.

The algorithm has been tested on fragmented characters, blur characters, smeared or broken characters and also on the image which contain graphics and overlapping character. It has been observed that in pre-processing of image when we apply different techniques to remove unwanted graphics, noise, and unwanted text, the accuracy of character recognition has been increased.

Image can be cropped, resized, rotated left or right, zoom in ,zoom out, and can also adjust resolution of image, if unwanted text, graphics will be removed from image extra noise in the sample image will be reduced, our studies shows that if pre-processing of image is applied properly the accuracy of character detection is improved at great extent. All the module of pre-processing has been developed in Java Integrated Environment (Java IDE).

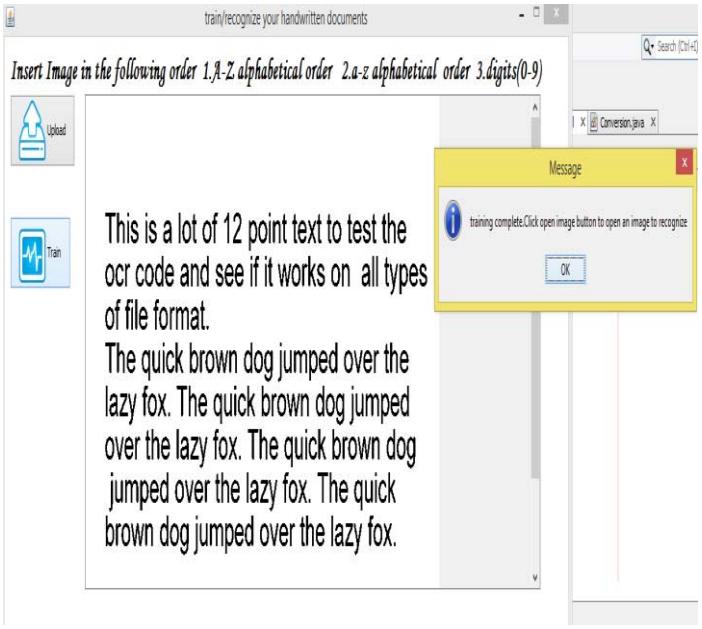

Fig. 8. Training of Data Set

Fig 9. Shows the training of neural network, after training the two dimensional array[x][y] get stored in the mysql database. This stored bit pattern matrix has been used further as the output node of neural network for the recognition of character.

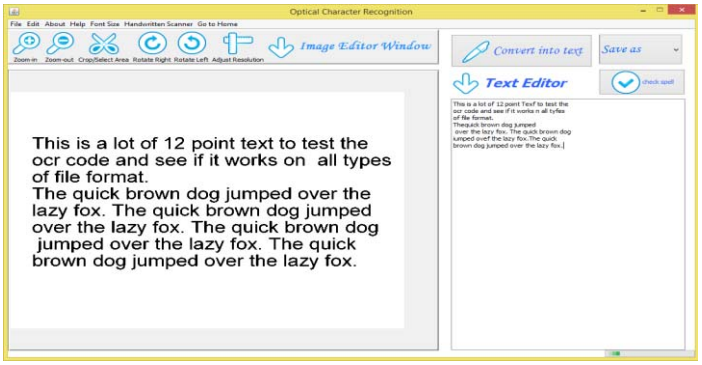

Fig. 9. Text Recognition

Fig 10. Show the recognition phase of character recognition, in this phase the sample image has been taken, in the text Editor and the recognised data has been stored, for the post processing of image

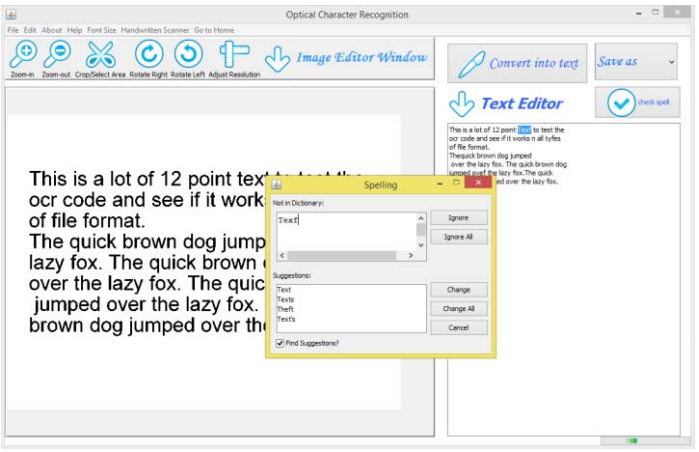

Fig. 10. Post Processing Through Dictionary

Fig 11. Shows the post processing of image by the use of dictionary the unrecognised letters produces spelling mistake of word, this words are further passed to dictionary for the correction of words.

Table II Accuracy of proposed algorithm

No of characters	Character recognised	accuracy
400	369	92.3%
560	506	90.5%
221	202	91.7%
322	275	85.4%

X. CONCLUSION

In this paper we have discussed complete methodology to implement character recognition, it has been observed that template matching a feature extraction method when applied with ANN (artificial neural network) i.e. the back propagation despite of low convergence rate of back propagation and pattern dependency offer several advantages in pattern recognition rate, we observed that feature extraction is probably the single most important factor for gaining high accuracy. Moreover the pre-processing methods discussed in this paper also remove noise at great extent. The paper discuss about the methodology used in character recognition application develop in java, so that it can easily run on heterogeneous networked computing platform.

XI. REFERENCES

[1] M. Cai, J. Song and M. R. Lyu. A New Approach for Video T ext Detection. In Proc. of International Confer ence On ImageProcessing, Rochester, New York,USA, pp. 117-120, 2002.

[2] Ching Y. Suen and Robert J. Shillman, "Low Error Rate Optical Character Recognition of Unconstrained Handprinted Letters Based on a Model of Human Perception", IEEE Transactions on Systems, Man, and Cybernetics, June 1977.

[3] S.N.Srihari, Y.C.Shin, V.Ramprasad and D.S.Lee. "Name and Address Block Reader" Proceedings of IEEE.

[4] R. Lienhart and W. Effelsberg. Automatic T ext Segmentation and T ext Recognition for Video Indexing. Multimedia System, V ol. 8, pp. 69-81, 2000.

[5] Wong, E. K. and M. Chen, "A robust algorithm for text extraction in color video", Proceedings of IEEE Int.Conference on Multimedia and Expo (ICME2000), 2000.

[6] Charif12, Omar, Hichem Omrani, and Philippe Trigano. "OMBP: Optic Modified BackPropagation training algorithm for fast convergence of Feedforward Neural Network." (2011).

[7]. A. Rodriguez, S. Kim, J. H. Ki, Y. B. Fernandez,"English To Spanish Translation Of Signboard Images From Mobile Phone Camera", IEEE Southeastcon '09 Pp. 356–361, 2009.

[8] Durai, S. Anna, and E. Anna Saro. "Image compression with back-propagation neural network using cumulative distribution function." World Academy of Science, Engineering and Technology 17 (2006): 60-64.

[9] D. Sasikala, R. Neelaveni, "Correlation CoefficientMeasure of Multimodal Brain Image Registration using Fast Walsh Hadamard Transform", Journal of Theoretical and Applied Information Technology, 2005.

[10] D. Sasikala, R. Neelaveni, "Correlation CoefficientMeasure of Multimodal Brain Image Registration using Fast Walsh Hadamard Transform", Journal of Theoretical and Applied Information Technology, 2005.

[11] N.K.Bose,P.Liang "Neural Network Fudnamentals With Graphs,Algorithms and Applications"

[12] V. P. Riyaz , T Truong."A System for Automated Data Entry from Forms", 1996 IEEE
Proceedings of ICPR '96

[13] R. Smith. "An overview of the Tesseract OCR Engine." Proc 9th Int. Conf. on Document Analysis and Recognition, IEEE, Curitiba, Brazil, Sep 2007, pp629-633.

[14] W.Wilson Lian,Heuristic-Based OCR Post-Correction for Smart Phone Applications the university of North Carolina at chapel hill department of computer science honors thesis Author 2009.

[15] U. Pal, M. Mitra, B.B. Chaudhuri, Multi-skew detection of Indian script documents, in: Proceedings of the Sixth International Conference on Document Analysis and Recognition, 2001, pp. 292–296.

[16] U. Pal, S. Datta, Segmentation of Bangla unconstrained handwritten text, in: Proceedings of the Seventh International Conference on Document Analysis and Recognition, 2003, pp. 1128–1132.

[17] H. Byan, S.W. Lee, A survey on pattern recognition applications of support vector machines, Int. J. Pattern Recognition Artif. Intell. 17 (2003) 459–486.

[18] Gllavata,Julinda, R.Ewerth,B Freisleben. "A robust algorithm for text detection in images." Image and Signal Processing and Analysis, 2003. ISPA 2003. Proceedings of the 3rd International Symposium on. Vol. 2. IEEE, 2003.

[19] Eikvil, Line. "OCR-optical character recognition." (1993).