**Optical character recognition of Handwritten and Printed English Text**

Umair muneer butt1, Umer maqsood1, Shahid bashir1, Nouman dilawar1, Muhammad umar1

Department of computer sciences Gift University Gujranwala Pakistan1

Emails:{umairmuneerbutt,u3sn.soft,shahidbashir,noumandilawar}@gmail.com

# Abstract:

The paper describes the development of two different types of OCR i.e. handwritten text and printed text. In optical character recognition the text image is processed through five phases. Bothe the OCRs for hand written and for printed text involve these five phases however the techniques involved in these phases differ. In pre-processing phase OCR convert RGB image to binary image and remove paper noise, color noise, background noise. Segmentation phase for printed text is done through connected components and for handwritten text the density based segments and connected component techniques are used. In feature extraction phase the structure of characters is analyzed. In classification both OCR use neural network that is a big challenge using it in mobile application. For hand written text recognition trained handwritten characters were trained and for printed text the samples for many fonts with different sizes were trained. In post processing the words are arranged into order. For handwritten text around 70% accuracy was achieved and for printed text accuracy reached up to 90%.

# Keywords:

OCR, opencv, ROI, RGB, paper noise.

# 1 Introduction:

Optical character recognition (OCR) is a program that translates scanned or printed image document into a text document. For certain language script today, it is not difficult to develop an optical character recognition (OCR) [6] system that recognizes well-Shaped characters with accuracy of 99% and above. In order to recognize the text contained in a document, it is usually segmented into lines, words, and characters. The typical phases of an OCR system are [7]:

Pre-processing

Segmentation

Feature extraction

Classification

Post-processing

Both handwritten and printed characters can be recognized, but the performance is directly dependent upon the quality of the input image.The simpler the input is to the trained samples, the better the performance of the OCR system will be. However, when it comes to totally unconstrained handwriting, OCR machines are still a long way from reading as well as humans. However, the computer reads fast and technical advances are continually bringing the technology closer to its ideal.

This section describes the basic OCR functions and in the preceding section the proposed methdology for the OCR is elaaborated. Pre-processing is the first phase in which image is smooth, blured so that noise can be removed. Image can also be gray scaled so that noise can be removed and image becomes clear .



Figure 1: Colored Image



Figure 2: Binarized Image

Image is then binarized and converted into black and white image using threshold as shown in Figure2.

The next step is segmentation in which binarized image is segmented verticaly and horizantaly so that line and words are separated as shown in figure figure7 and figure 8. As in case of printed text segmentation is an easy task beacause writing is straight and there is space between words and characters so segmentation process becomes esay,there are different approaches to separate words, eg.vertical segmentation, connected components etc. Segmenting handwritten english text is a challenging task beacause there is no hard and fast rule to segment it, everyone has his own writing style so it is difficult task to segment the handwritten text. We have used density based segments to separate the characters from the words as indicated in figure 9.

Th third stage is feature extraction in which words’ features are extracted to determine if it is a character or any object or some time the feature are extracted by matching techniques as shown in table 1. These features are used for character recognition.

In the classification phase, the extracted words are classified to already defined classes like numbers, alphabets. For classification purpose there are a number of techniques, for example, template matching and neural networks can be used to obtain good result . For tempalte matching sample image are kept as sample and match with extracted words and classify them as shown in figure 11, it is showing the pixel matching process. In case of neural network [8], there are feed forward and feed backward networks which are superb techniques for classification purpose.

In post processing phase words are arranged and put on a file from a image. In the following sections proposed methdology is presented for an OCR system for mobile device that works for printed and handwritten english text and numbers.

## 2 Selected Methodology

After a complete survey of OCR phases the following methodology was adopted:

## 2.1 Pre-processing

The first stage in OCR phases is pre-processing which takes raw images and applies the operations as given below:

### 2.1.1 Binarization

A binary image is a digital image that has only two possible values for each pixel. This means that each pixel is stored as a single bit (0 or 1), named black and white image as shown in figure 4. Open CV 2.3.1 was used [1] and the function getAdaptivelocalbinarization ()for Binarization was implemented. But it was noted that in case of printed document this technique is not quite efficient as there are different density images. Different formulas on different images were used to calculate the percentage the following for formula which is a variation upon Otsu’s work [5] was selected:

**threshold = (average - ( 1+(int) ( 0.2 \* ( (standerdDeviation / R ) -1 )) ))%245-standerdDeviation (1)**

It works well and gives 95% accuracy in case of printed text. For handwritten text Open CV function work well. The parameters were set as: **getadaptivelocalbinarization (17, 14, image), w**here 17 are the window size and 14 is the constant that check the pixel value from a point to 14 pixels in possible directions.

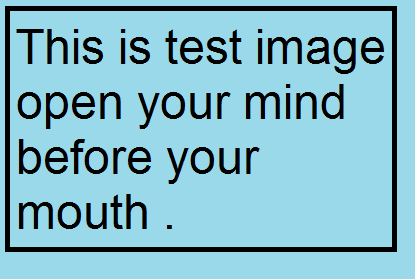


Figure 3: Input Image

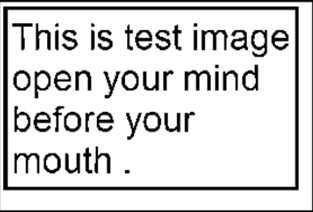


Figure 4: Binarized Image

### 2.1.2 Noise Reduction

In Binarization some of the noise is removed but some noise remains generally in the image. The dilation process was opted to remove noise with the help of connected component and the length and width of black bodies were noted and neglect them in the image as shown in the figure 5. The size of the noise was calculated its length and width and noise was removed.

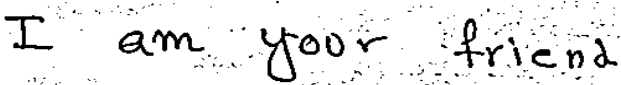


Figure 5: Noisy Image Before and After the Noise Reduction

## 2.2 Segmentation

Segmentation phase consists of following steps:

### 2.2.1 External Segmentation

For printed text Segmentation is naturally not a challenging task but in case of handwritten text it is difficult to separate the words. The proposed method works on discrete handwritten text and does not work very well on joining writing. Common which are as follows: First of all Region of interest (ROI) of the image is extracted as shown in the figure 6 required area is surrounded by green and red regions.

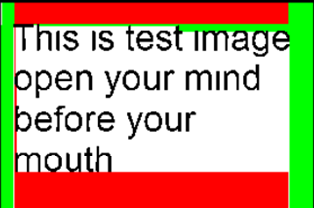


Figure 6: Region of Interest

After that those lines are separated as shown in figure 7 red line are showing that line is eliminated from one point to end.

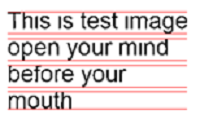


Figure 7: Lines Segmentation

Words are separated using vertical segmentation due to some space between them as shown in figure 8.

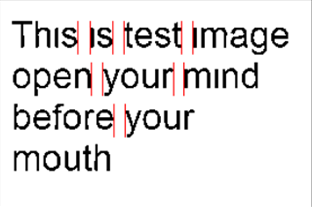


Figure 8: Words Segmentation

### Internal Segmentation

In internal segmentation each character is extracted from each world. For joining handwritten text segmentation [4] base line and thickness base technique was used to segment them and then connected component [2] to eliminate joins as shown in the figure 9 indicated by red lines. Connected component approach then perform reasonably well on both printed and handwritten text as shown in figure 10.



Figure 9: Joining handwritten letter Segmentation

C:\Users\Umair\Desktop\Untitled.png

Figure 10: Segmentation using connected components with character labeled with different colors

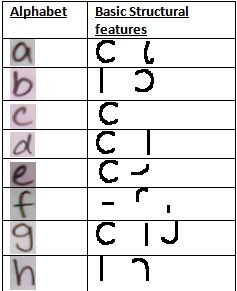
## Feature Extraction

Feature extraction is a technique by which image analysis is done .an image has different kind of feature we will discuss some of as follows:

### 2.3.1 Structural Features

To recognize characters the shape of the characters and their structure were analyzed and it was found quite helpful but did not give desired accuracy. Here are some features as shown in table one, character a is composed of two ligatures, b is composed of two and g is composed of 3

Table 1: English alphabets structure analysis



## 2.4 Classification

In classification the characters obtain from image were classified to already defined classes like A,B,C etc. For classification purpose different techniques for matching were used. First one is pixel by pixel matching.

### 2.4.1 Pixel Matching

Pixel matching [3] was experimented and was found quite useful for the printed text but did not give suitable result for handwritten. As shown in figure 11 Q is an image which is converted in o and 1 and then a black image is formed which is matched to trained image



Figure 11: Pixel Matching Process

### 2.4.2 Basic Shapes Matching

Structural features of the characters are one of the techniques that help a lot. Some data structure to store the shapes was used and the best fit was checked. Again it worked best for printed because shapes are definite but not for handwritten as shown in the figure 12 input image is matched with ligatures in the database with unique indexes.

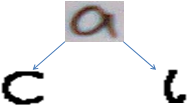


Figure 12: Matching Technique of Structural Feature

### 2.4.3 Neural Networks

Classification performance is given by neural networks [7] that worked best for both printed and handwritten. Multi-Layer perceptron feed forward network was used for the purpose and neuroph library was exploited to train a network which contain a training element which is supervised

**Training Set<Supervised Training Element >(288, 62**).

This is a training element that was used in training process. 288 are bits or pixel of the image and 62 are neurons that were used in training process.

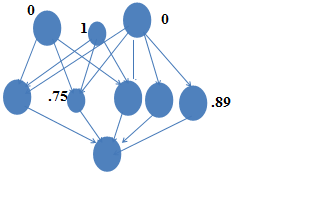


Figure 13: Multilayer Perceptron Feed Forward Network

Transfer function sigmoidis used, set error rate .001 and maximum iterations 500. The image was given as input in it as in the form of 0 and 1 and defined output was obtained. Neural network was a big challenge it involves certain steps to form a network are as follows:

### 2.4.4 Collecting Samples

First phase to train a network is to collect samples from different people in case of handwriting as shown in figure 14 and in printed samples from different writing styles were collected. The **times new roman** font was used in the project. In case of handwriting, the neural network with 17\*62 = 1054 alphabet samples was trained. The handwritten samples from 17 peoples were collected which contained 62 characters and the neural network was trained for these samples.

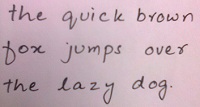


Figure 14: Sample of writing to be Train

### 2.4.5 Analysis of Samples

After collecting the samples they were refined and then prepared for training as shows in figure 15. These refined images gave better result than a noisy image.

Figure 15: Cropped Samples



### 2.4.6 Rescaling

After cropping the images were rescales according to image and up-scaling and downsizing of small and large images respectively was done to match the size of the trained samples.

### 2.4.7 Vectored Samples

AS a computer knows the language of 0 and 1’s, the image were converted into vector of 0 and 1 to train as shown in figure 16.

Figure 16: Samples converted in to matrix form



Then the vectored image was trained to MLP. After all training, a trained .nnet file was obtained which was used then in OCR.

## 2.5 Post-processing

In post processing the noise was removed from the image and the segmented characters were rearranged as they were in the text image.

# 3 Results

In the context of above techniques, methods and different strategies we are to able to compile results and the preceding section will describe the different techniques impact on results. As image is going through certain stages then we got its accuracy. I will describe in shortly the process as described before as this is a mobile application image is captured from a certain angle from any place then it is input to the algorithm that first grey scaled image then binaries him. For printed text we have our own method of Binarization and for handwritten we have used opencv. Image is then segmented across horizontal and vertical axis. At character level image is rescaled so that matching techniques could perform better. For training purpose we have used multi layer perceptron that is feed forward and supervised. We trained the network with alphabets and test them on others. We got 90 percent or above accuracy on printed English text and around 70 percent accuracy on handwritten text. Results are given below with details.

### 3.1 Result Analysis of Printed English Text

English language consists of different font styles with different sizes. We mainly work on time new roman, Calibri and Arial. We trained our neural network on some font sizes and calculate the accuracy which can be enhanced more by more training and more font styles. These results are gathered on applying certain constrains and different kind of printed images which are as follows.

Table 2: Accuracy of Printed Text on Different Fonts and Styles

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Document Type** | | **Samples** | **Characters** | | | **Character Accuracy** | **Words** | | **Words Accuracy** |
| Font-Style | Image Type | No. of pages | Total | recognized |  |  | Total | recognized |  |
| Arial | Neat & cleaned | 10 | 190 | 164 |  | 86.3% | 36 | 23 | 58.3% |
|  | Normal | 10 | 205 | 171 |  | 83.4% | 46 | 22 | 47.8% |
|  | Noisy | 10 | 197 | 149 |  | 75.6% | 39 | 17 | 43.5% |
| Times  New | Neat & cleaned | 10 | 252 | 216 |  | 85.7% | 51 | 30 | 58.8% |
| Roman | Normal | 10 | 300 | 254 |  | 84.6% | 58 | 27 | 46.5% |
|  | Noisy | 10 | 198 | 152 |  | 76.7% | 40 | 17 | 42.5% |
| Calibri | Neat & cleaned | 10 | 213 | 172 |  | 80.3% | 42 | 21 | 52.3% |
|  | Normal | 10 | 243 | 179 |  | 73.6% | 49 | 20 | 40.8% |
|  | Noisy | 10 | 298 | 216 |  | 72.4% | 61 | 24 | 39.3.% |

## 3.2 Result Analysis of Handwritten English Text

For handwritten English text we took samples from different individuals. Some people writing were discrete and some make joining. We basically work on discrete and on joining we have done a little work which can be improved in future. These are the results on discrete handwritten text.

Table 3: Accuracy of Handwritten text

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Alphabets | | Number of samples | Neural Network | Test samples | | Recognition rate | |
| Multi layer perceptron (FF),Neurons=288, | Trained  sample | Test sample | Trained  Sample | Untrained  Sample |
| Discrete alphanumeric characters | a-z ,A-Z and 0-9 | 2000 characters | epochs=1000 | 1200 | 800 | 90% | 60% |
| Discrete alphanumeric words | a-z, A-Z and 0-9 | 200 words  (15 pages) | epochs=1000 | 120 | 80 | 75% | 55% |

# 4 Discussions

Optical character recognition field depend mainly on results that how much accuracy you have achieved. The best your matching result results with the original text the best your OCR will be. In case of printed English text if we specify our domain OCR can improve up result 99% .As there are a number of fonts with different fonts it is not easy to recognize well. Now a day’s artificial intelligence that deals with machine learning so that we can learn certain styles and test it and it can give 100 per cent results. But as everything has limitations we cannot learn all the fonts to the algorithm. OCR field improve a lot and used in document scanning and other purposes.

In this case printed text result on random text images result are above 90 %.so we can achieve better result if we define our domain. Results can goes to 100 per cent, it depends upon the image if it’s clear not so much noise then results will be better some assumptions like lightning condition, image capturing angle etc. will be consider . There are certain things that cannot be recognized correctly like number 1, capital i, small i and small L. Similarly number 0, small o, capital o are not recognized correctly because there is no as such difference between them. So these are some you can say challenges or bugs in character recognition. In case of handwritten English text accuracy is yet a big challenge because there are lot of writing variations so we can work on a hard fast rule; everyone has its own style. So OCR for handwritten is not as accurate as printed text. Handwritten text can be categorized into 2 categories discrete and joining. Discrete letters can easily be processed and give accuracy up to 90 per cent. Joining letters segmentation is a big challenge because in segmentation process some characters shape is changed and cannot recognized correctly you can see in Table 4 and 5 above.

# 5 Conclusions

The Application recognizes English words and digits and displays them on the screen with reasonable accuracy.so in this experiment; the proposed algorithm is tested with several document images. Some of the documents contained overlapping lines and characters. Even though it could segment all the documents in a robust way and gave good results. But, it couldn’t segment the touching lines and characters. The broken characters have been over segmented. Segmentation of the touching lines and characters may require some heuristic approaches.in case of printed text we achieve 90% and above high accuracy we remove background nice like images as we have to extract text.

## 6 References

[1]. Sajjad ,K.M.: Automatic License Plate Recognition using Python and Opencv, Department of Computer Science and Engineering,M.E.S. College of Engineering, Kuttippuram, Kerala.

[2]. Rakhmadi, A., Othman, N. Z. S., Bade, A., Shafry, M., Rahim, M., Amin, I. M.: connected component labeling using components neighbors-scan labeling, Journal of Computer Science 6 (10): 1096-1104 (2010).

[3]. Teshome, M., Zerubabel, L., Yoon, K.D.: A Simple Binary Image Similarity Matching Method Based On Exact Pixel Matching, International Conference on Computer Engineering and Applications IPCSIT vol.2(2011)IACSIT Press, Singapore.

[4]. Yuan, A., Bai, G., Yang, P., Guo, Y., Zhao, X.: Handwritten English Word Recognition based on Convolution Neural Networks, International Conference on Frontiers in Handwriting Recognition(2012).

[5]. OTSU, N.: A Threshold Selection Method from Gray-Level Histograms, IEEE TRANSACTIONS ON SYSTREMS, MAN, AND CYBERNETICS, VOL. SMC-9, NO. 1 (1979).

[6]. Alata, M., Al-Shabi, M.: TEXT DETECTION AND CHARACTER RECOGNITION USING FUZZY IMAGE PROCESSING, Journal of ELECTRICAL ENGINEERING, VOL. 57, NO. 5, 2006, 258–267.

[7]. Pradeep, J., Srinivasan, E., Himavathi, S.: 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).

[8]. Mandal , R.K., Manna,N.R.: Hand Written English Character Recognition using Row- wise Segmentation Technique (RST), *International Journal of Computer Applications.*