Performance Analysis of Time in Character Segmentation Using Different Fonts

Shashi Kant[1], Sini Shibu[2]
[1]*M.Tech Research Scholar, Department of CSE*
[2]*Asst. Professor, Department of CSE*
NRI Institute of Information Sc. & Tech., Bhopal

***Abstract*— Segmentation is a widely used process in current-day image processing. Various segmentation processes exist to segment characters and words. header lines are detected and converted as straight lines. Text line segmentation is an important step because inaccurately segmented text lines will cause errors in the recognition stage. Text characteristics can vary in font, size, orientation. This paper introduces a comparative analysis of the segmentation process that is carried out on various style text to find the time taken in the segmentation process. The text is segmented into lines, lines into words and then from lines words Here we make a comparative study of time consumed during character segmentation using different style fonts and find the time in which fonts type, character segmentation takes minimum amount of CPU time. Here we preposed a method that supports all segmentation process(character segmentation) to retrieve text, make boundary boxes to the characters and perform character segmentation. It consists of recording the start time of the process as well as end time of the process and find the time difference. This process is applied to different fonts of characters to make a comparative analysis.**

***Index Terms*- Image processing, Printed document analysis, Text font time analysis, segmentation, word segmentation and character segmentation.**

I. INTRODUCTION

Segmentation of a document image into its basic entities, namely, text lines words and characters, is considered as a non-trivial problem to solve in the field of printed document recognition. The difficulties that arise in printed documents make the segmentation procedure a challenging task. Different types of difficulties are encountered in the text line segmentation, word segmentation and characters procedure. In case of text line segmentation procedure, major difficulties include the difference in the skew angle between lines on the page or even along the same text line, overlapping words and adjacent text lines touching. Printed character recognition is difficult task compared to machine printed different fonts style character recognition in the area of Optical Character Recognition. A lot of research is done on the printed English text, but less work has been done on the printed English text characters recognition. It has four major steps such as pre-processing, segmentation recognition and CPU time consuming. Segmentation is the important step. Segmentation also contains three major steps such as line segmentation, word segmentation and character segmentation. If we fail in doing character segmentation then entire segmentation process goes wrong. No more research has been done in the past on characters segmentation of machine printed different fonts style texts. This approach basically limits the processing time accuracy achieved by the text recognition system.

Optical Character Recognition, usually shortened to OCR, is the electronic translation of scanned of typewritten or printed text images into machine-encoded/computer-readable text. The popularity of OCR has been increasing each year with the advent of fast microprocessors providing the base for vastly improved recognition techniques. Now, there have been tremendous improvements in increasing both effective read rates and accuracy of character recognition. Desktop OCR scanners can recite typewritten data into a computer at rates up to 2400 words per minute! As we all know it is used for many types of data entry whether bank statement, business card, passport documents, invoices, receipts, mail, or any number of printed records.

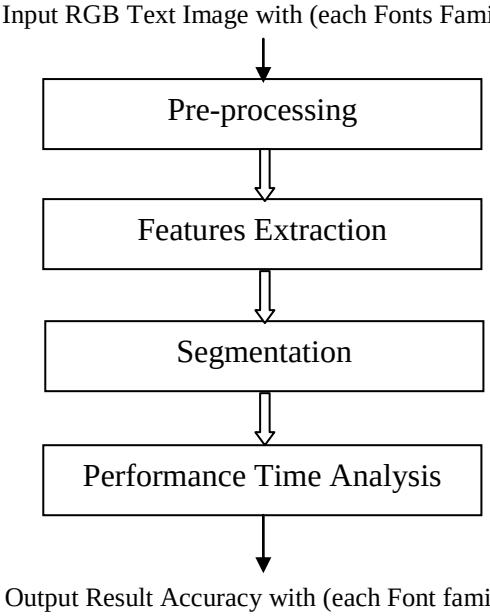

Fig. 1 Text Information Extraction System

II. SEGMENTATION BASIC

Segmentation is one of the most important phases in character recognition process. Segmentation is the process of segmenting the whole document image into recognizable units.

The segmentation process is divided into three types.
A. Line segmentation
B. Word segmentation
C. Character segmentation

A. Line Segmentation:

First step of segmentation process is segmenting the text region into lines, also called as line segmentation. First we need to calculate header lines and base lines for the Line segmentation Header lines are rows with maximum number of black pixels and base lines are rows with minimum number of black pixels. Finding header line is a challenge because of skew in headline.

B. Word Segmentation:

Word segmentation is easier than line segmentation and character segmentation. Space between two words is generally more than three pixels. Words are segmented by the projection based method.
Word segmentation is the problem of dividing a string of written language into its component words. In English and many other languages using some form of the Latin alphabet, the space is a good approximation of a word divider (word delimiter).
However the equivalent to this character is not found in all written scripts, and without it word segmentation is a difficult problem. Languages which do not have a trivial word segmentation process include Chinese, Japanese, where sentences but not words are delimited,

C. Character Segmentation:

Main problem in character segmentation is separating header line from word. In printed or scanned text header line is not straight line. But now to the process is being done on printed documents with different fonts families. However we performing the character segmentation on different fonts like Times New Roman, Arial Rounded, Calibri , verdana text, Cambria and Berlin sans text etc. by using sample text images.

The rest of this paper is organized as follows. Section III presents some of the related works in text localization, and segmentation. Section IV describes the implementation of the two segmentation approaches. Experiments and result analysis are discussed in Section V. Section VI concludes the paper with future recommendations.

III. LITERATURE REVIEW

Text detection is usually performed on text features such as vvvcolor, edge, texture. Text detection is classified into three main categories: Color based, Edge based and Texture based. Marcin Pazio et al. [5] used color based approach for text detection. Segmentation is followed by connected component analysis. Azadboni [11] proposed a two stage algorithm for text localisation. In first stage, image is preprocessed to remove noise and increase the contrast, followed by projection profile analysis to extract text blocks. In the second stage, extracted text blocks are verified using SVM classifier. Finally segmentation is performed to extract character pixels. Huang et al.[12] used Stroke Feature Transform (SFT) followed by a text component classifier and a text-line classifier, sequentially to extract text regions. Finally, text regions are located by thresholding the text-line confident map. In [6] [7] texture based approach is discussed in which text is considered as a specific form of texture. In [8][9][10], edge based approach for text detection is discussed in the literature. Zramdini et al. [1] introduced font style detection method by calculating the CPU time taken in the horizontal profile of the whole text block. R. K. Yadav et al. [2] described a simple and fast algorithm for detection of italic and bold characters in different fonts in Devnagari script, without recognition of the actual character. N. Sharma et al. [3] also presented a technique for improving the recognition accuracy of Hindi OCR system by developing the concept for detection of bvold, italic word of different fonts. It is evident from the literature survey that none of the researchers had focussed on detecting total time is to be taken in segmentation on different fonts based and all capital Fonts pattern words in printed in Arial Rounded, Times New Roman, etc script till date. This has motivated us to propose a two-stage font invariant detection technique for detecting all the above mention type font time accuracy during processing

IV. COLOR BASED AND FONTS BASED TEXT SEGMENTATION

This section discusses segmentation approach for text detection. In text, letters should have uniform color and fonts family within a text string. This property is used in color based text detection and fonts based time analysis approaches. The general algorithm for color based and font based character text detection approach is as follows. Initially Text image is segmented.
Step 1: As simply color image is processed and make transforming it into rgb2gray for extracting the color.
Step 2: Binarization the processed image and noise removing and bounding rectangular region on to the text into number of connected components.
Step 3: Perform Connected component analysis is performed on the bounded text image to find the character segments.
Step 4: Calculating CPU time, taken by the overall segmentation process on different fonts families.

Here we discuss some texts image machine printed or scan RBG text image with different types of fonts namely-arial,times new roman, calibri etc in Fig. 2. As we kno that for the time performance and analysis during segmentation to find the overall fonts based segmentation process performed in less time to optimize the result. For this process a routine process is to be as follows.

A. Pre-processing.
B. Feature Detection
C. Result Analysis.

A. Pre-processing:

In pre processing the printed or scan text image is converted into ready to use format or the noise in the image is reduced by using these methods, in the scanned image the noise

would be like character shapes are not accurately scanned document was not associated properly, the document is tilted to rare degrees, edges are not smooth, multiple colored characters, So, the remove such noise, we have to apply some logics as computer doesn't know the difference between the noise and the accurate character. To remove the noise the following algorithms can be applied to make it ready to use for character recognition.

De-skew:

This technique is used, If the document was not aligned properly when scanned, it may need to be tilted a few degrees clockwise or anti-clockwise in order to make lines of text perfectly horizontal or vertical.

Binarization:

This technique is used to convert an image from RGB color to grayscale to black-and-white to make it appropriate for character recognition and it is also known as "binary image" because there are only two colors left in the image after the binarization i.e. "BLACK" and "WHITE".

B. Feature Detection:

After the Binarization of the 2D text image in to binary pixels values (0, 1). The rectangular box is to formed in to each of characters to reduce to one or more character prototypes. And this is also known as intelligent character recognition (ICR) though this is much more sophisticated way of spotting characters. Most omni font OCR programs i.e. ones that can recognize printed text in any font work by feature detection rather than pattern recognition.

C. Result Analysis:

Segmentation of printed or scanned text from image consists of different fonts and there expected process time analysis of each is briefly discussed in V phase.

Fig. 2. Illustration of six fonts family text namely- Arial Rounded, Cambria, Calibri, Verdana, Times New Roman and Berlin sans.

V. RESULT AND DISCUSSION

The performance analysis of time during different fonts text segmentation approaches in character detection and segment one by one, is carried out in a set of 12 text images selected from printed or scanned documents. The selected images contain both images with uniform color intensity and non uniform intensity. The algorithms are implemented in MATLAB R2010a. A subjective evaluation is performed on the segmentation results. Table 1. shows the output of segmentation from different font type text where all containing words are processed to find more than 500 characters to be segmented correctly and arise some errors during process and produced resultant output to find the accuracy of 92.3 % efficiently the Times new Roman based segmentation. In RGB based segmentation, changes in intensity causes part ofcharacters to merge into background. Time analysis of this existence proposed method is also by a graphical form in Fig. 3. below.

Different Font Family	Character segmentation performance analysis				
	Text word	**Total characters**	**Correctly Segmented**	**Error**	**Time (ms.) Accuracy**
Arial Rounded	100	500	97.2 %	2.8 %	91.8 %
Times New Roman	100	500	97.5 %	2.5 %	92.3 %
Cambria	100	500	96.4 %	3.6 %	88.9 %
Calibri	100	500	95.4 %	4.6 %	87.2 %
Verdana	100	500	96.8 %	3.2 %	89.5 %
Berlin Sans	100	500	94.3 %	5.7 %	86.4 %

Table 1. Output Table with Result.

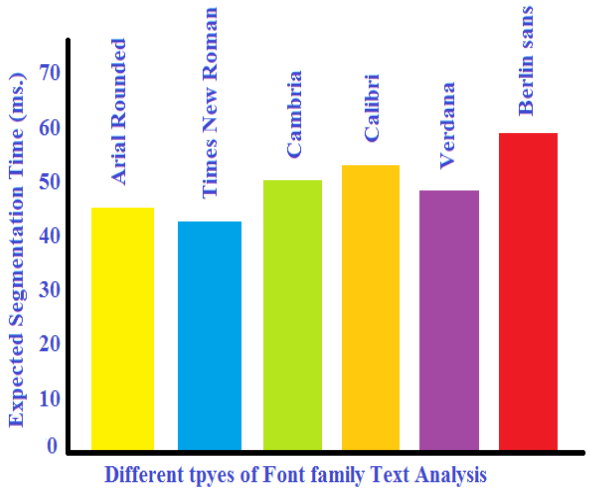

Fig. 3. Analysis of time taken by each font.

VI. CONCLUSION

From the above table we can say that Times New Roman and Arial Rounded fonts are easier to read and detect for character segmentation. This may take less time and make effective segmentation on text-lines. As compared to all the different fonts text character techniques we found that every font consist of different variance and orientation. We implement the code in MATLAB and perform font based text segmentation. Thus, in the proposed method we found

that segmentation is most efficient technique with Times New Roman and Ariel Rounded fonts..

REFFERENCES

[1] A. ZramdiniS, R. Ingold, "Optical font recognition using typographical features," In: IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, pp. 877-882, 1998.
[2] R. K. Yadav, B. D. Mazumdar, "Detection of Bold and Italic Character in Devanagari Script", In: International Journal of Computer Applications, vol. 39, no. 2, pp. 19-22, 2012.
[3] N. Sharma, M. Khandelwal, "Detection of Bold Italic and Underline Fonts for Hindi OCR", In: International Journal of Computer Trends and Technology (IJCTT), vol. 4, Issue 8, pp. 2425-2428, 2013.
[4] R. C. Gonzalez, R. E. Woods, "Digital Image Processing", vol. I. Prentice-Hall, India (1992).
[5] Marcin Pazio, Maciej Nied´zwiecki, Ryszard Kowalik, Jacek Lebied´z, "Text Detection System for the Blind", 15th European Signal Processing Conference (EUSIPCO 2007), Poznan, Poland, September 3-7, 2007,pp.272-276.
[6] S. A. Angadi & M. M. Kodabagi, "Text Region Extraction from Low Resolution Natural Scene Images using Texture Features", International Journal of Image Processing, vol. 3, issue 5, pp. 229-245, 2009.
[7] Kwang In Kim, Keechul Jung, And Jin Hyung Kim ,"Texture-Based Approach For Text Detection In Images Using Support Vector Machines And Continuously Adaptive Mean Shift Algorithm" , IEEE Transactions On Pattern Analysis And Machine Intelligence , Vol. 25, No. 12, 2003.
[8] Manoj Kumar , Gueesang Lee, "Automatic Text Location from Complex Natural Scene Images" The 2nd International Conference on Computer and Automation Engineering (ICCAE) 2010 IEEE vol. 3, pp. 594-597.
[9] Andrej Ikica, Peter Peer, "An improved edge profile based method for text detection in images of natural scenes", EUROCON-International Conference on Computer as a tool (EUROCON) 2011 IEEE: 1-4.
[10] Jing Zhang and Rangachar Kasturi, "Text Detection Using Edge Gradient and Graph Spectrum",2010 International Conference on Pattern Recognition, 2010 IEEE ,pp 3979-3982.
[11] Azadboni, M.K. ; Behrad, A."Text detection and character extraction in color images using FFT domain filtering and SVM classification", 2012 Sixth International Symposium on Telecommunications (IST), 2012 IEEE ,pp 794-799. DOI: 10.1109/ISTEL.2012.6483094
[12] Weilin Huang, Zhe Lin , Jianchao Yang ,Jue Wang, "Text Localization in Natural Images Using Stroke Feature Transform and Text Covariance Descriptors",International Conference on Computer vision (ICCV), 2013 IEEE ,pp 1241-1248.