International Journal of Computer Applications Technology and Research
Volume 3– Issue 4, 239 - 243, 2014, ISSN: 2319–8656

Optical Character Recognition from Text Image

Ranjan Jana
Department of MCA
RCC Institute of Information Technology
Kolkata, India

Amrita Roy Chowdhury
Department of MCA
Academy of Technology
Hooghly, India

Mazharul Islam
Department of CSE
RCC Institute of Information Technology
Kolkata, India

Abstract: Optical Character Recognition (OCR) is a system that provides a full alphanumeric recognition of printed or handwritten characters by simply scanning the text image. OCR system interprets the printed or handwritten characters image and converts it into corresponding editable text document. The text image is divided into regions by isolating each line, then individual characters with spaces. After character extraction, the texture and topological features like corner points, features of different regions, ratio of character area and convex area of all characters of text image are calculated. Previously features of each uppercase and lowercase letter, digit, and symbols are stored as a template. Based on the texture and topological features, the system recognizes the exact character using feature matching between the extracted character and the template of all characters as a measure of similarity.

Keywords: character recognition; feature extraction; feature matching; text extraction; character extraction

1. INTRODUCTION

Optical character recognition (OCR) is the conversion of scanned images of printed, handwritten or typewritten text into machine-encoded text. This technology allows to automatically recognizing characters through an optical mechanism. In case of human beings, our eyes are optical mechanism. The image seen by eyes is input for brain. OCR is a technology that functions like human ability of reading. OCR is not able to compete with human reading capabilities. OCR is a technology that enables you to convert different types of documents such as scanned paper documents, PDF files or images captured by a digital camera into editable and searchable data.

One widely known application is in banking, where OCR is used to process demand draft or cheque without human involvement. An image of demand draft or cheque can be captured by mobile camera, the writing on it is scanned instantly, and the correct amount of money is transferred. This technology has nearly been perfected for printed demand draft or cheque, and is fairly accurate for handwritten demand draft or cheque as well, though it requires signature verification. In the legal industry, there has also been a significant movement to digitize paper documents. In order to save space and eliminate the need to sift through boxes of paper files, documents are being scanned. OCR further simplifies the process by making documents text-searchable, so that they are easier to locate and work with once in the database. Legal professionals now have fast, easy access to a huge library of documents in electronic format, which they can find simply by typing in a few keywords. OCR is widely used in many other fields, including education, finance, and government agencies.

In this paper, one effective optical character recognition from text image using texture and topological features is proposed. For better performance, the texture and topological features of all characters of text image like corner points, features of different regions, and ratio of character area and convex area are calculated. Based on the texture and topological information, character verification is done using feature matching between the extracted character and the template of all character serves as a measure of similarity between the two. This paper is organized into the following sections. Section II describes an overview of previous work. Implementation details for optical character recognition are mentioned in section III. Experimented results are shown in section IV. Finally, the conclusions are in section V.

2. PREVIOUS WORK

Several approaches for text detection in images and videos have been proposed in the past. Based on the methods being used to localize text regions, these approaches can be categorized into two main classes: connected component based methods and texture based methods. The first class of approaches employs connected component analysis, which consists of analyzing the geometrical arrangement of edges or homogeneous color and grayscale components that belong to characters. The second class of approaches regards texts as regions with distinct textural properties, such as character components that contrast with the background and at the same time exhibit a periodic horizontal intensity variation, due to the horizontal alignment of characters.

An automatic text extraction system is proposed in [1], where second order derivatives of Gaussian filters followed by several non-linear transformations are used for a texture segmentation process. Then, features are computed to form a feature vector for each pixel from the filtered images in order to classify them into text or non-text pixels. Methods of texture analysis like Gabor filtering and spatial variance are used to automatically locate text regions in [2]. A new approach is proposed in [3] to perform a color reduction by bit dropping and color clustering quantization, and afterwards, a multi-value image decomposition algorithm is applied to decompose the input image into multiple foreground and background images. An approach in which LCQ (Local Color Quantization) is performed for each color separately is proposed in [4]. Each color is assumed as a text color without knowing whether it is real text color or not. [5] has presented an algorithm which uses only the red part of the RGB color space to obtain high contrast edges for the frequent text colors. By means of a convolution process with specific masks it first enhances the image and then detects edges. [6] has presented a technique that performs an eight-connected component analysis on a binary image, which is obtained as the union of local edged maps that are produced by applying the band Deriche filter on each color. A work on chinese

script recognition for business card images is reported in [7]. A new approach for video text detection is reported in [8].

A number of research works on mobile OCR systems have been found. Motorola China Research Center have presented camera based mobile OCR systems for camera phones in [9]. A business card image is first down sampled to estimate the skew angle. Then the text regions are skew corrected by that angle and binarized thereafter. Such text regions are segmented into lines and characters, and subsequently passed to an OCR engine for recognition. The OCR engine is designed as a two layer template based classifier. A similar system is presented for Chinese-English mixed script business card images in [10]. An outline of a prototype Kanji OCR for recognizing machine printed Japanese texts and translating them into English is proposed in [11]. An approach of character recognition system for Chinese scripts has been presented in [12]. A system is developed for only English capital letters in [13]. At first, the captured image is skew corrected by looking for a line having the highest number of consecutive white pixels and by maximizing the given alignment criterion. Then, the image is segmented based on X-Y Tree decomposition and recognized by measuring Manhattan distance based similarity for a set of centroid to boundary features. However, this work addresses only the English capital letters and the accuracy obtained is not satisfactory for real life applications. Moreover, research in developing OCR systems for mobile devices is not limited to document images only. [14] worked on reading LCD/LED displays with a camera phone. Text/Graphics Separation for Business Card Images for Mobile Devices is proposed in [15]. A Fast Skew Correction Technique for Camera Captured Business Card Images is proposed in [16]. Segmentation of Camera Captured Business Card Images for Mobile Devices is proposed in [17]. Optical character recognition still remains an open challenge for many languages.

3. IMPLEMENTATION

Optical character recognition (OCR) takes a text image as input and gives editable text document as output. The OCR system primarily involves four steps: Pre-processing, Features extraction, Features training, and Feature matching. Flow chart of the OCR is shown in Figure 1. Here, two data sets are considered, one for training dataset and another for test dataset. Preprocessing and feature extraction is done in both cases. Features extracted from test data is compared with features extracted from training data to get the desired output.

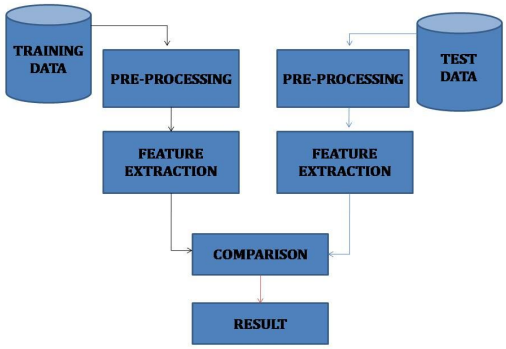

Figure 1: Flowchart of OCR system

3.1 Pre-processing

The text image is converted into binary image for further working as shown in Figure 2 and Figure 3. When any image is converted to binary, it is easy to work with pixel values 0 and 1. The binary image is complimented so that the letters constitute by binary 1 (one) and background constitute by binary 0 (zero) as shown in Figure 4.

Figure 2: Text Image

Figure 3: Binary Image

Now, individual text lines are separated from the binary image. This is done by calculating the sum of all values in a row. When the sum is 0, a new line is identified and separation is done. The sum of all rows in between two lines should be zero. The image is divided into several lines and each line is extracted one by one as shown in Figure 5. This procedure is repeated until all lines are extracted.

Figure 4: Complimented Binary Image

Figure 5: Extracted lines

Single lines are extracted due to the fact that, dealing with one line is easier than dealing with the whole image. Again, for each line, the letters are to be extracted as shown in Figure 6 and Figure 7. This is done by calculating the sum of all values in a column. When sum is zero, a character is identified and separation is done. In this way, all individual characters (alphabets, digits, punctuations) are separated.

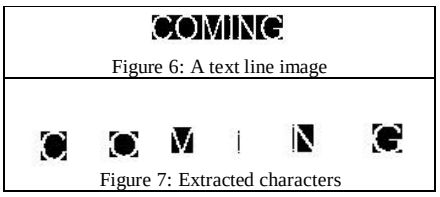

Figure 6: A text line image

Figure 7: Extracted characters

3.2 Features Extraction

Feature extraction technique is applied for all individual extracted characters. The character image is divided into four regions as shown in Figure 8.

Figure 8: Extracted character

Sum of the pixels value of the whole image and sum of pixels value in each of the sub-regions are calculated. Then their ratios are calculated as the features value of f1, f2, f3, f4 respectively.

f1= Sum of the pixels value of 1st quadrant /
Sum of the pixels value of the whole image

f2= Sum of the pixels value of 2nd quadrant /
Sum of the pixels value of the whole image

f3= Sum of the pixels value of 3rd quadrant /
Sum of the pixels value of the whole image

f4= Sum of the pixels value of 4th quadrant /
Sum of the pixels value of the whole image

To get better accuracy, features f5, f6, f7, f8, f9, and f10 are calculated using f1, f2, f3, and f4.

f5=f1+f2

f6=f2+f3

f7=f3+f4

f8=f1+f4

f9=f2+f4

f10=f1+f3

Using Harris corner method, numbers of corner points are calculated from character image. Feature f11 is considered as the number of corner points of a character. Total area of extracted character image is calculated using the actual number of pixels in the character image. Convex area of the character is calculated using the number of pixels in convex hull that can contain the character region. Feature f12 is ratio of convex area to total area.

f12= Convex Area / Total Area

Total twelve features f1 to f12 are extracted for all individual extracted characters.

3.3 Features Training

Here, three fonts, namely 'Lucida Fax', 'Berlin Sans' and 'Arial' have been considered as training data set. Three images Figure 9, Figure 10, and Figure 11 is used to extract the character features for training the system. The trained features value will be used for recognizing the extracted character.

Figure 9: Text Image of Lucida Fax

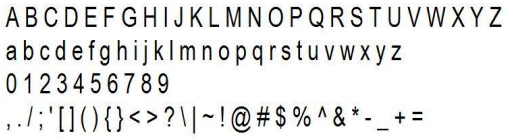

Figure 10: Text Image of Arial

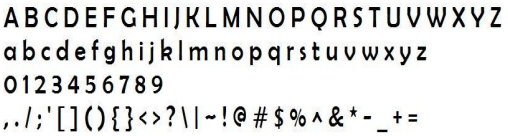

Figure 11: Text Image of Berlin Sans

3.4 Feature Matching

The features value is matched with the trained features set to recognize the exact character. Different matching algorithm can be used for feature matching. The minimum distance value with respect to all the features (f1 to f12) is selected as required character.

ALGORITHM FOR OCR

STEP 1: The input text image is converted into binary image.

STEP 2: The binary image is complimented so that the letters constitute by binary 1 (one) and background constitute by binary 0 (zero).

STEP 3: All text lines are separated from the binary image. This is done by finding the sum of all values in a row. When the sum is 0, a new line is identified and separation is done. The sum of all rows in between two lines should be zero.

STEP 4: For each line, the characters are to be extracted. This is done by finding the sum of all pixels value in a column. When sum is zero, a new character is identified and separation is done.

STEP 5: Total 12 features value f1 to f12 are extracted for each character.

STEP 6: The features value are matched with the trained features set to recognize the exact character.

4. EXPERIMENTAL RESULTS

This section introduces the experimental results. Only three fonts, namely 'Arial', 'Berlin Sans' and 'Lucida Fax' have been considered as training data set. Based on training data set, five cases each comprising of different text images (different number of characters) with five different fonts is tested for optical character recognition. The letters which are correctly and incorrectly interpreted are counted and the accuracy is calculated as shown in Table 1.

Table 1: Result of OCR

Test Image	Font Name	Correct Recognition	Incorrect Recognition	Accuracy
Case 1 (consist of 10 characters)	Arial	6	4	60%
	Berlin Sans	10	0	100%
	Cambria	5	5	50%
	Lucida Fax	6	4	60%
	Times New Roman	6	4	60%
Case 2 (consist of 20 characters)	Arial	19	1	95%
	Berlin Sans	19	1	95%
	Cambria	6	14	30%
	Lucida Fax	16	4	80%
	Times New Roman	7	13	35%
Case 3 (consist of 25 characters)	Arial	24	1	96%
	Berlin Sans	25	0	100%
	Cambria	1	24	4%
	Lucida Fax	14	11	56%
	Times New Roman	7	18	28%
Case 4 (consist of 30 characters)	Arial	28	2	93.33%
	Berlin Sans	30	0	100%
	Cambria	5	25	16.66%
	Lucida Fax	20	10	66.66%
	Times New Roman	9	21	30%
Case 5 (consist of 70 characters)	Arial	61	7	87.14%
	Berlin Sans	66	4	94.28%
	Cambria	10	60	14.28%
	Lucida Fax	45	25	64.28%
	Times New Roman	15	55	21.42%

The text image of Arial font is used as input of the OCR as shown in Figure 12. The output of OCR is the editable text document as shown in Figure13. Out of 70 characters, 61 characters are correctly interpreted.

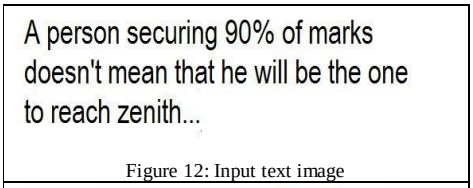

Figure 12: Input text image

A pers8n securing 9M% 8fmarks
d8esn"t mean that hewill bethe 8ne
t8 reach zenith...

Figure 13: Output of OCR

5. CONCLUSIONS

A number of methods have been proposed by several authors for optical character recognition. A new method to extract features from text images and recognition of exact character to produce text document is presented here. The proposed method promises a very simple but reliable solution to the problem of optical character recognition. The technique that is used based on calculating the number of corner points and utilizing the various properties like object area and convex areas of the image. Only three fonts, namely 'Arial', 'Berlin Sans' and 'Lucida Fax' have been considered as training data set. Experimental results on a set of images show accuracy up to 100% for 'Berlin Sans', 96% for 'Arial' and 80% for 'Lucida Fax'. Achieved results are encouraging and suggest the adequacy of the selected features. Accuracy for 'Cambria' and 'Times New Roman' font is very poor. Accuracy for these fonts can be achieved by training the system with 'Cambria'

and 'Times New Roman' characters set. The proposed algorithm will help the community in the field of handwritten character recognition. By introducing more features, the accuracy can be enhanced.

6. ACKNOWLEDGEMENT

The authors are grateful to Anirban Dasgupta, Anindita Dey, and Ankita Kumari for their help and support to improve this paper. The authors are also grateful to all the faculty members of MCA department, RCC Institute of Information Technology, Kolkata for providing us constant support and facilities.

7. REFERENCES

[1] V. Wu, R. Manmatha and E. M. Riseman, " Finding Text in Images", In Proc. of Second ACM International Conference on Digital Libraries, Philadelphia, PA, pp. 23-26, 1997.

[2] H. Li and D. Doermann, "Automatic Text Tracking In Digital Videos", In Proc. of IEEE 1998 Workshop on Multimedia Signal Processing, Redondo Beach, California, USA, pp. 21-26, 1998.

[3] A. K. Jain and B. Yu, "Automatic Text Location in Images and Video Frames", In Proc. of International Conference of Pattern Recognition (ICPR), Brisbane, pp. 1497-1499, 1998.

[4] P. K. Kim, "Automatic Text Location in Complex Color Images Using Local Color Quantization", IEEE TENCON, Vol. 1, pp. 629-632, 1999.

[5] L. Agnihotri and N. Dimitrova, "Text Detection for Video Analysis", In Proc. of the International Conference on Multimedia Computing and Systems, Florence, Italy, pp. 109-113, 1999.

[6] C. Garcia and X. Apostolidis, "Text Detection and Segmentation in Complex Color Images", In Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP2000), Istanbul, Vol. 4, pp. 2326- 2330, 2000.

[7] W. Pan, J. Jin, G. Shi, Q. R. Wang, "A System for Automatic Chinese Business Card Recognition", ICDAR, pp. 577-581, 2001.

[8] M. Cai, J. Song and M. R. Lyu, "A New Approach for Video Text Detection", In Proc. of International Conference On Image Processing, Rochester, New York, USA, pp. 117-120, 2002.

[9] X. Luo, J. Li and L. Zhen, "Design and implementation of a card reader based on build-in camera", International Conference on Pattern Recognition, pp. 417-420, 2004.

[10] X. Luo, L. Zhen, G. Peng, J. Li and B. Xiao, "Camera based mixed-lingual card reader for mobile device", International Conference on Document Analysis and Recognition, pp. 665-669, 2005.

[11] M. Koga, R. Mine, T. Kameyama, T. Takahashi, M. Yamazaki and T. Yamaguchi, "Camera-based Kanji OCR for Mobile-phones: Practical Issues", Proceedings of the Eighth International Conference on Document Analysis and Recognition, pp. 635-639, 2005.

[12] K. S. Bae, K. K. Kim, Y. G. Chung and W. P. Yu, "Character Recognition System for Cellular Phone with Camera", Proceedings of the 29th Annual International Computer Software and Applications Conference, vol. 1, pp. 539-544, 2005.

[13] M. Laine and O. S. Nevalainen, "A standalone OCR system for mobile camera-phones", Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium, pp.1-5, Sept. 2006.

[14] H. Shen and J. Coughlan, "Reading LCD/LED Displays with a Camera Cell Phone", Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop, 2006.

[15] A. F. Mollah, S. Basu, M. Nasipuri and D. K. Basu, "Text/Graphics Separation for Business Card Images for Mobile Devices", Proc. of the Eighth IAPR International Workshop on Graphics Recognition (GREC`09), pp. 263-270, July, 2009, France.

[16] A. F. Mollah, S. Basu, N. Das, R. Sarkar, M. Nasipuri, M. Kundu, "A Fast Skew Correction Technique for Camera Captured Business Card Images", Proc. of IEEE INDICON-2009, pp. 629-632, 18-20 December, Gandhinagar, Gujrat.

[17] A. F. Mollah, S. Basu, M. Nasipuri, "Segmentation of Camera Captured Business Card Images for Mobile Devices", International Journal of Computer Science and Applications, 1(1), pp. 33-37, June 2010.