Text Extraction from Natural Scene Images using Prewitt Edge Detection Method

Volume 3, Issue 12, December 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Text Extraction from Natural Scene Images using Prewitt Edge Detection Method

Prof. Amit Choksi[1], Nihar Desai[2], Ajay Chauhan[3], Vishal Revdiwala[4], Prof. Kaushal Patel[5]

Electronics and Telecommunication Department, BVM Engineering College,
Anand, India

***Abstract**— Digital images captured by camera contains many useful information. Many research has been done for content retrieval from natural images, for example, object detection, face recognition and automatic text extraction. In recent years, many methods are suggested for extracting text data from images as it has wide range of applications. Three popular algorithms for this are: Edge detection, Connected Component and texture-based. The purpose of this paper is to compare the two basic methods Prewitt Edge Detector and Gaussian Edge Detector with different structuring elements. The algorithms are implemented and applied to an image set with different text size, font styles and text language. Performance is evaluated based on the precision rate and recall rate for each method on the same image set.*

***Keywords**— Character extraction, Text Region Detection, Edge Detection, Text Localization, Prewitt operator*

I. INTRODUCTION

Recent studies in the field of research [15],[16] on the content retrieval from images and videos identified a wide variety of applications that require automated systems for text extraction. One such recently developed application is the mobile banking application provided by the banking institutions that facilitates the customers to carry out the transactions even on passing the image of the cheque to the server. All other such applications include tourist guide which facilitate the tourists to understand the display boards though they are unfamiliar with the local language of that place and Image text translation systems to help the visually impaired people and also tourists. Every such application relies on a Textual Information Extraction (TIE) system which can efficiently detect, localize and extract the text information present in the natural images.

Text data is particularly interesting, because text can be used to easily and clearly describe the contents of an image [2]. A variety of applications are found in recent studies that uses extracted text. One such recently developed application is the mobile banking application provided by the banking institutions that facilitates the customers to carry out the transactions even on passing the image of the cheque to the server [6]. A translation camera is another application that captures the images, detect the text from it and then translate it in required language from the local language. This can also be useful for visually impaired people if text to speech program (TTS) is used within Text Detection and Localization Recognition (OCR) [10],[11].

Input image Text presented in the image Research on text detection and localization is carried out since 1990s and numerous text detection algorithms have been proposed .All these approaches are majorly classified into three categories edge-based, connected-component based and texture based techniques. This paper describes the three techniques and implements them to compare by evaluating the performance. The rest of the paper is organized as follows: The detailed survey related to the various methods of text extraction in natural scenes is described in section 2, Approach for the implementation of three techniques is presented in section 3, Experimental results of performance analysis is presented in section 4, Section 5 concludes the work and presents the future recommendations.

II. RELATED WORK

Many algorithms have been proposed for recognizing text data in an image. Each method gives robust results for specified set of images. However, results may be varying due to image size, orientation, contrast, and colour. In non-document images, detecting text is more challenging because of variation in text size and alignment. The simple block diagram for text information extracting system is shown in the figure 1.

Input image is any image that contains text information. The first step detects the regions where the text is present. Text localization refers to locate and enhance text areas. Character extraction refers to eliminate non-text regions from image, such that output image contains only text. This image is sent to OCR system to obtain the characters that are extracted.

Edge-based text extraction

As mentioned previously, edge-based algorithm is one of the popular schemes for text extraction. Edge-based methods focus on the 'high contrast between the text and the background' and the edges of the text boundary are identified and merged [2],[6]. In this paper, edge-detector based method is proposed and compared with conventional Gaussian pyramid [1] method. Comparison is done in terms of precision rate and recall rate.

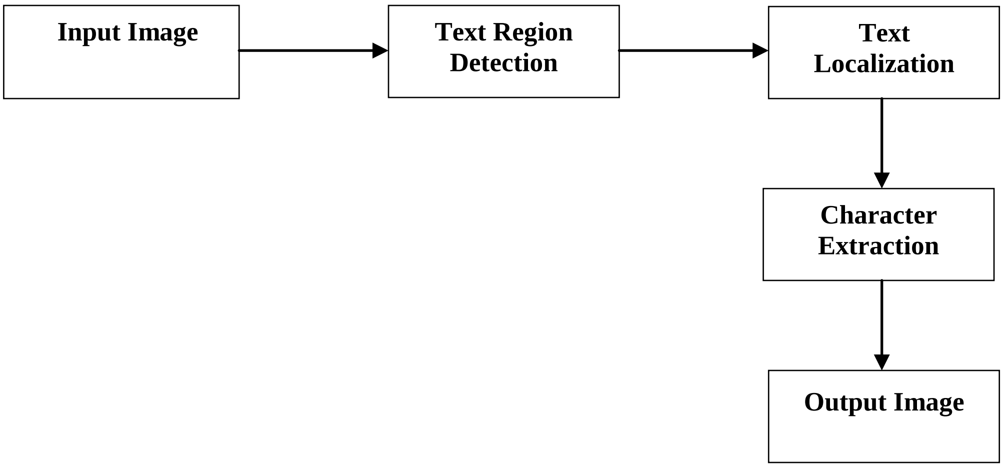


Fig.1Procedure of Text Extraction from Image

A. *Algorithm that uses Gaussian pyramid*
1. Create a Gaussian pyramid by convolving the input image with a Gaussian kernel and successively down-sample each direction by half. (Levels: 4)
2. Create directional kernels to detect edges at 0, 45, 90 and 135 orientations.
3. Convolve each image in the Gaussian pyramid with each orientation filter.
4. Combine the results of step 3 to create the Feature Map.
5. Dilate the resultant image using a sufficiently large structuring element (7x7) to cluster candidate text regions together.
6. Create final output image with text in white pixels against a plain black background.

B. *Algorithm that uses Prewitt edge-detector*
1. Convert the image into monochrome image by thresholding.
2. Filter the image for removing noise. Use Gaussian low-pass filter.
3. Apply Prewitt edge-detector to the filtered image.
4. Apply proper morphological operations, i.e. dilation to make clusters of text regions.
5. Multiply the resultant image with input black and white image to get text in contrast with Plain background.

III. APPROACH

A. *Pre-processing*

Pre-processing steps are necessary to improve the performance and make the process efficient to the time. This includes gray-scaling and binarization of image and filtering to remove noise.

1. Gray-scaling: The given image is multicolor RGB image, in which text may not be separated from the background. In color image, each pixel is combination of R (Red), G (Green) and B (Blue) and values varying from 0 to 255. For gray-scaling, these values are added in a proportion of Red: 30%, Green: 59% and Blue: 11% to get the gray scaled [3] equivalent of that particular pixel.
2. Binarization: This converts gray-scale image into binary image i.e. containing only Black (0) and White (1) pixels. Gray-scaling gives a threshold for binarization of image. To be specific, this is done by comparing each pixel value to a threshold value (that lies between black and white) and setting that pixel value to black or white as its consequence [3]. This process noticeably separate (or distinguish) text from image background.
3. Filtering: Any image taken from camera contained noise such as blurred image, high frequency noise and white noise. To improve image quality and for further processing on image, Gaussian low pass filter is used. It has following properties: 1) Gaussian smoothing is very effective for removing Gaussian noise. 2) They are linear low pass filters. 3) Rotationally symmetric (perform the same in all directions). 4) The degree of smoothing is controlled by σ (larger σ for more intensive smoothing)

   In our case, it is important to remove white noise, while maintaining salient edges because text contains edges. This can be a contradictory task-white noise exists at all frequencies equally, while edges exist in the high frequency range. (Sudden changes in spatial signals-text regions). Gaussian LPF is best-suited for this because it has a graceful and natural tail that becomes ever lower as the frequency increases. This means that it will act as a low pass filter, but also allow in higher frequency components matching with how quickly its tail decays.

B. *Edge-Detection*

Edges are those places in an image that correspond to object boundaries. Edges are pixels where image brightness changes abruptly. Specifically in text data probably more edges are present than non-text areas. For example, Letters 'E', 'Z', 'H','A' etc. are having horizontal and/or vertical edges. If we detect these edges, there may be likelihood of other letters or words around (because words are usually grouped) Thus, the text region is detected. We chose Prewitt amongst several edge-detectors available like Sobel, canny and Roberts. Choice of Prewitt is quite empirical. Prewitt edge-detector detects horizontal and vertical edges in an image and combines them to give resultant image.

C. *Morphological Operations*

After detecting text region(s), a cluster of it is created such that the all letters are covered. Morphological dilation is used for this purpose as dilation adds pixels to the boundaries of objects in an image thereby thickening that object. Measure of thickness is defined by the type and size structuring element. Proper sized structuring element should be chosen such that least non-text area should be clustered within. Here, structuring element 'disk' with size 9 (a disk of radius 9) is used.

To remove non-text objects significantly, morphological opening operation is used. Opening operation is erosion followed by dilation. It is performed to remove objects of specific size from image. This size is again determined by structuring element. After performing such operations, the resultant image holds clusters of text regions having pixel value 1 (white).

D. *Character extraction*

This step refers to identify the characters as they are in original image. This is done by multiplying resultant image with binary converted original image. In this operation, pixels having value 1 (i.e. text) are recovered as same in original image and pixels having value 0 are present as background. However, the final image may contain some non-text part, extent of which is measured by precision rate. Final result is the white text in black background or vice versa, dependent on the original image.

IV. EXPERIMENTAL RESULTS & ANALYSIS

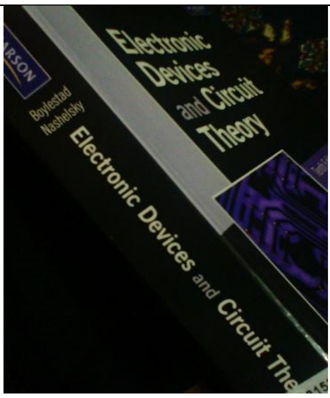
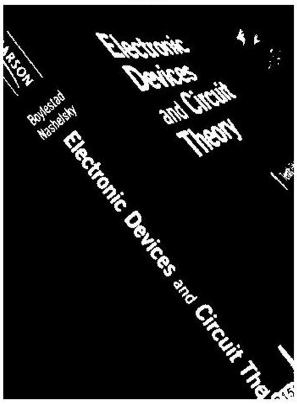
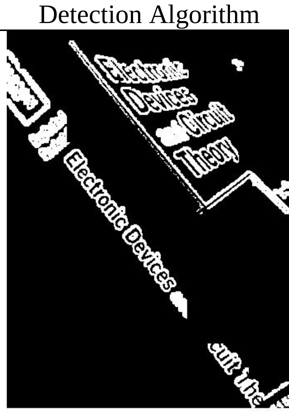
Performance of each algorithm are evaluated and compared in terms of precision rate [1] and recall rate [1]. Evaluation is done over 30 test images with different text size, colour contrast and orientation. Results are summarized in table 1 and 2. Precision and recall rates are calculated as follows:

$$\text{Precision Rate} = \frac{Correctly detected words}{Correctly detected words \quad + False positives} X100\% \tag{1}$$

$$\text{Recall Rate} = \frac{Correctly\ detected words}{Correctly detected words \quad + False negatives} X100\% \tag{2}$$

False positives are the non-text regions in the image and have been detected by the algorithm as text regions. False negatives are the text regions in the image and have not been detected by the algorithm. Table 1 shows the input images and their output results using Prewitt edge detection and Gaussian edge detection algorithm. Here it has been seen that text information is easily extracted using Prewitt edge detector as compared with Gaussian detector. The algorithm is also very successful to remove non text regions from the images as compared with Gaussian edge detector. The recall rate and precision rate of various natural scene images are shown in table 2.

TABLE I RESULT IMAGES OF TEXT DETECTION METHODS

Images	Output of Prewitt Edge Detection Algorithm	Output of Gaussian Edge Detection Algorithm
		

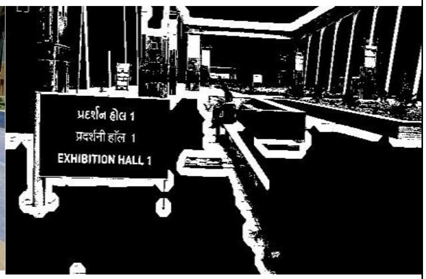
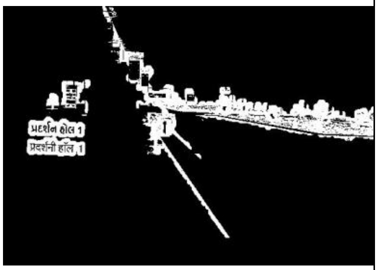





TABLE II PRECISSION RATE AND RECALL RATE OF VARIOUS NATURAL SCENE IMAGES

	Precision Rate		**Recall Rate**	
Image	**Prewitt (%)**	**Gaussian (%)**	**Prewitt (%)**	**Gaussian (%)**
Cover.jpg	91.6	90	84.61	69.23
Exhibit.jpg	75	85.79	100	66.66
Station.jpg	71.42	83.3	100	100
msu.jpg	**69.23**	**0**	**90**	**0**
Book.jpg	**100**	**100**	**100**	**53.84**
Efy.jpg	93.1	95.5	90	76.66
Dept.jpg	28.5	0	15	0
Smoke.jpg	76.41	80	65	56.52
Rupee.jpg	**89.47**	**66.66**	**54.83**	**6.66**

From above table 2 it is cleared that precision rate and recall rate using Prewitt edge detection is almost very high as compared with Gaussian edge detection method.

V. CONCLUSION

Most of researchers have shown that edge detection method is not successfully satisfied for text detection form natural scene images, but here we have proved that edge detection using Prewitt edge detector gives almost same result compared with connected component algorithm. This algorithm is also tested on more than 50 images and got satisfied result. Here the results obtained by Prewitt and Gaussian edge detection algorithm on different set of images are been compared with respect to precision rate and recall rate. The Edge based method detects text regions with highest Precision (100 %) & Recall (100 %) rate. Thus it is more efficient compared to that of the performance obtained with Gaussian edge detection method. We observed that Gaussian edge detection based methods is not good enough to detect the text regions which is shown in table 1. In order to add with that Prewitt edge-detector algorithm is more robust since it gives higher recall rate as compared Gaussian edge-detector algorithm. The edge based algorithm is also able to give better results in case of lighting variance compare to Gaussian edge detection method and also on illuminated images.

REFERENCES

[1]. Sharma S.[2007], "Extraction of Text Natural Images".
[2]. Giri, P. S.[2013], "Text Information Extraction And analysis from Image Using Digital Image Processing Techniques"
[3]. Agarwal P, Varma R [2012] “Text Extraction From Images”, IJCSET
[4]. Ferreira S., Garin V.,Gosselin B.,"A Text Detection Technique Applied in the Framework of a Mobile Camera-Based Application".
[5]. Gonzalez R.C., Woods R.E., Eddins S.L.[2004], Digital Image Processing Using MATLAB, Pearson Education.

[6]. M. Swamy Das, B. HimaBindhu, A. Govardhan[2012], "Evaluation of Text Detection and Localization Methods in Natural Images."
[7]. Jung K, Kim KI, Jain AK,[2004] "Text Information Extraction in Images and Video"
[8]. Angadi, S.A. and Kodabagi, M.M, Text region extraction from low resolution natural scene images using texture features, 2ndInternational Advance Computing Conference, IEEE,2010.
[9]. Xiaoqing Liu and Jagath Samarabandu, the University of Western Ontario, Department of Electrical & Computer Engineering.
[10]. Angadi, S.A. and Kodabagi, M.M, Text region extraction from low resolution natural scene images using texture features, 2ndInternational Advance Computing Conference , IEEE,2010.
[11]. Xiaoqing Liu and Jagath Samarabandu, An Edge-based text region extraction algorithm for Indoor mobile robot navigation, Proceedings of the IEEE, July 2005.
[12]. Xiaoqing Liu and Jagath Samarabandu, Multiscale edge-based Text extraction from Complex images, IEEE, 2006.
[13]. Julinda Gllavata, Ralph Ewerth and Bernd Freisleben, A Robust algorithm for Text Detection in images, Proceedings of the 3rd international symposium on Image and Signal Processing and Analysis, 2003.
[14]. Anoual, H. and Aboutajdine, D. and Elfkihi, S. and Jilbab, A, Features extraction for text detection and localization, I/V Communications and Mobile Network (ISVC)
[15]. Sushma, J. and Padmaja, M. Intelligent Agent & Multi-Agent Systems, Text detection in color images, 2009, IAMA 2009.
[16]. Partio, M. and Cramariuc, B.and Gabbouj, M. and Visa, A, *Rock* texture retrieval using gray level co-occurrence matrix, Proc. of 5th Nordic Signal Processing Symposium.
[17]. Shehzad Muhammad Hanif and Lionel Prevost, Texture based Text Detection in Natural Scene Image: A help to blind and visually impaired persons, Conference & Workshop on Assistive Technologies for People with Vision & Hearing Impairments Assistive Technology for All Ages CVHI 2007, M.A. Hersh (ed.).
[18]. C.P. Sumanthi, T. Santhanam and N.Priya**,** Techniques and Challenges of automatic Text Extraction in Complex Images: A Survey, Journal of Theoretical and Applied Information Technology, january 2012 .
[19]. Xiaoqing Liu and Jagath Samarabandu, A Simple and Fast Text Localization Algorithm for Indoor Mobile Robot Navigation, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 5672, 2005.
[20]. Qixiang Ye, Qingming Huang, Wen Gao and Debin Zhao, Fast and Robust text detection in images and video frames, Image and Vision Computing 23, 2005.
[21]. Qixiang Ye, Wen Gao, Weiqiang Wang and Wei Zeng, A Robust Text Detection Algorithm in Images and Video Frames, IEEE, 2003.