STM JOURNALS

Journal of Artificial Intelligence Research & Advances

Volume 1, Issue 2
www.stmjournals.com

Text Extraction from Images Using Connected Component Method

Arvind*, Mohamed Rafi
Computer Science and Engineering Department, U.B.D.T College of Engineering, Visvesvaraya Technological University, Davangere, Karnataka, India

Abstract

Text information from image serves as an important clue for many image based applications. it has been used in many different applications such as; document analysis, vehicle licence plate extraction, content based information indexing, To name a few, the detection and extraction of text regions in images is a well-known problem in computer vision and image processing. However, variation of these texts due to differences in size, orientation style, and alignment, as well as low image contrast and complex background make problem of automatic text detection and extraction tremendously difficult and challenging one. Here the connected component method applied on the dataset which includes the images which categorized according to the language of the scene text captured.

Keywords: *Text Information Extraction (TIE), Text Detection, Connected Component Method.*

**Author for Correspondence* E-mail: arvindbelure@gmail.com

INTRODUCTION

The text has a significant role in video and images as it is an important parameter used for retrieval system and also for the information system. Automatic detection and extraction of text from images or videos is an important problem in many applications like document processing, image indexing, video content summary, factory automation, wearable computers, and since including. The stages of a text extraction system are presented in Figure 1, and it generally has two phases.

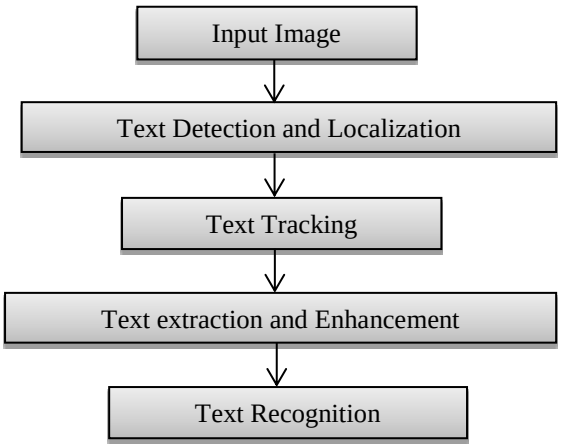

Fig. 1: *Text Extraction System.*

The detection and localization of the text in first step whereas, the Recognition is second step, where the detected text regions are fed into OCR which recognizes the characters and gives the textual output.

The complete explanation of each steps are described below:

Text Detection

It is the process of determining the presence of text in a given image/frame (normally text detection is used for a sequence of images).

Text Localization

It is the process of determining the location of text in the image and generating bounding boxes around the text.

Text Tracking

Text tracking is carried out for reducing the processing time for text localization, although the location of text in an image can be shown by bounding boxes, the text still needs to be segmented from the background. The extracted text image is converted to binary image for processing into an OCR.

Text Extraction and Enhancement
In this step the text components are segmented from its background. Because of these text region commonly have a low resolution and it is susceptible to noise, these segmented components are enhanced for processing into OCR.

Recognition
By using OCR technology, the extracted text images can be transformed into plain text. The text recognition takes input as raw image, its main aim is to decide if there is some text present in the image and if so, to give output as the parts of image containing text, determine the exact co-ordinates of the text positions and find the meaning of the extracted text.

Text in images can be mainly classified into caption text (superimposed/artificial text), and scene text (graphics text) [1, 2].

- Text that is naturally exists in an image, and typically it does not represent anything important related to the content of the image. These types of texts are known as scene text.
- Text produced separately from the image is in general a very good key to understand the image. These types of texts are known as artificial text.

The scene text is more difficult to detect and very less amount of work has been done in this area. And compared to caption text, scene text can have any orientation. Furthermore, it is often affected by the camera parameters such as focus, motion, illumination and also affected by variations in scene.

Texts in the images generally have diverse appearance changes like font, color, style, orientation, alignment, contrast, texture, size, and background. All the alterations will make the problem of automatic text extraction more challenging. Text in images can show numerous variations as per the follow-on properties (Table 1) [1]:

Geometry

- ✓ Alignment: All characters in the text seem to be in groups and commonly have horizontal orientation, because of their distinct effects, sometimes they can look as non-planar texts. Scene text can be aligned in any direction and they can have geometric alterations.
- ✓ Size: The size of the text varies lot, so assumptions are made based on the domain of application.
- ✓ Inter-character distance: characters in the text line include a uniform separation between them.

Color
The text line characters have a tendency to have the same or comparable colors. Because of this property, a connected component-based method for text detection is to be used. Most of the works are done on single monochrome texts (text of single color).

Motion
The similar characters typically exist in successive video frames with movement or without a movement and this property is used in text tracking and enhancement stages of TIE. Caption text generally moves in a constant way (horizontally or vertically) whereas, scene text can have random motion due to the movement of camera or the object.

Edge
Most of the caption text and scene text are designed to be read easily, and hence resulting in the robust edges at the borders of text and background.

Compression
Various digital images are recorded, transferred, and processed in a compressed format. Therefore, a faster Text Information Extraction system can be attained if it can easily extract text without decompression.

BACKGROUND

Numerous methods for text detection in images have been offered previously. According to the methods used; they are categorized into different methods, such as; Connected component based method, Edge based methods, Region based method, texture based method and Mathematical morphology based method.

Table 1: *Properties of Text in Images.*

Property	**Variants or sub classes**	
Geometry	Size	Consistency in size of text
	Alignment	Straight line with skew (implies vertical direction)
		Curves
		3D perspective distortion
	Inter-Character distance	Aggression of characters with uniform distance
Color	Gray	
	Color (monochrome, polychrome)	
Motion	Static	
	Linear Movement	
	2D rigid constrained movement	
	3D rigid constrained movement	
	Free movement	
Edge	Strong edges (contrast) at text boundaries	
Compression	Un-compressed image	
	JPEG, MPEG- Compressed image	

Lienhart and Effelsberg [2] presented a method which operates directly on color images using the RGB color space. The character features like monochromacity and contrast within the local environment are used to qualify a pixel as a part of a connected component or not, segmenting each frame into suitable objects in this way. Then, regions are merged using the criteria of having similar color. Finally, specific ranges of width, height, width-to-height ratio and compactness of characters are used to discard all non-character regions.

Lee and Kankanhalli [3] proposed a method based on connected component for the detection and recognition of text from the image on the cargo containers, in which they have different lighting conditions and characters with many different shapes and sizes. Edge information formed is utilized for an abrasive search earlier to the generation of connected components. To determine the boundaries the difference between the adjacent pixels are used. Based on the pixels on the boundaries, the local thresholding values are selected. These characters are then used for generating the Connected components with the identical gray-level. Then, non-text components based on measurements of aspect ratio, histograms are filtered out. In spite of that they states that this method can be used in different areas.

Jain and Y u [4] presented in their work in which they performed a color reduction technique by bit dropping and color clustering, and then, to decompose the input image into multiple images, a multi-value image decomposition algorithm is applied. Later, connected component method is combined with the projection profile features are implemented on all of those to localize text region. By this approach they extracted only horizontal texts of large sizes in images.

Garcia and Apostolidis [5] proposed work in which an eight-connected component method, in that they applied a band Deriche filter on every color and got local edged maps and a binary image is obtained by combining edge maps.

Cai, et al. [6] presented work based on edge strength, edge density horizontal distribution properties. By color edge detection algorithm removed the not-text edges with the help of low thresholding. Later by local thresholding technique, the background is simplified and kept low-contrast text. Finally, text regions are localized by analysing the projection profiles.

DESIGN OF THE SYSTEM/PROPOSED METHOD

The basic block diagram of connected component method is shown in the Figure 2.

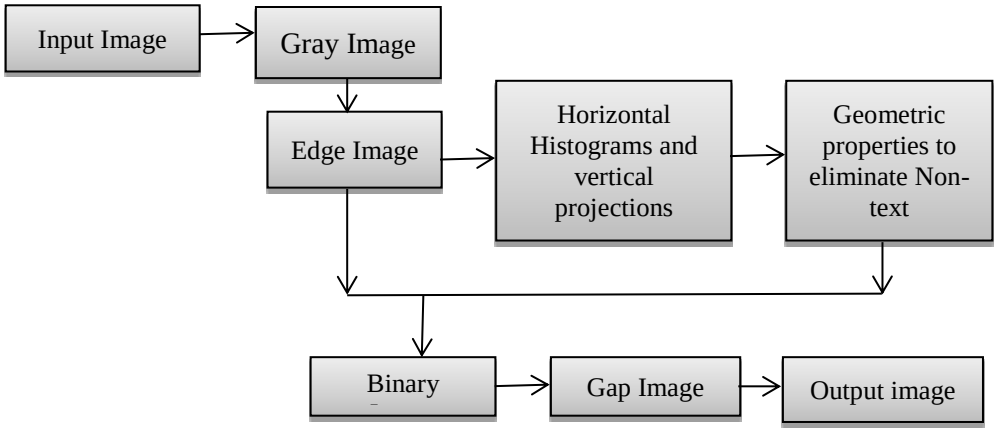

Fig. 2: *Basic Block Diagram of Connected Component Method.*

The proposed connected component algorithm [7] is as shown below:

Step 1: Convert the input image into YUV color space. The luminance(Y) value is further processed. The output is gray image.

Step 2: Convert the gray image into Edge image.

Step 3: Compute the horizontal and vertical projection profiles of a candidate text regions using a histogram with appropriate threshold value.

Step 4: Use the geometric properties of the text such as width-to-height ratio of characters to eliminate possible non-text regions.

Step 5: Binarize the edge image enhancing only the text region against plain black background.

Step 6: Create the gap image using gap filling process and use this as reference to further eliminate the non-text region from the output.

Step 7: Generate the text blobs.

Pre-Processing

The pre-processing is done for the easy detection of the text regions. As proposed in [7], the image is converted to YUV color space (luminance + chrominance) and only luminance(Y) channel is used for further processing as it is more clear. This conversion is made using MATLAB function 'rgb2ycbcr'. The individual channels can be extracted from new image. The Y channel denotes intensity or brightness of the image, however, channels U and V refers to the real information of color. Since text present in an image has more contrast with its background, by using only Y channel, the image can be converted to gray-scale image with only the brightness/contrast information present.

Detection of Edges

In this process, the connected-component based approach is used to make possible text regions stand out as compared to non-text regions. Every pixel in the edge image is assigned a weight with respect to its neighbours in each direction. As depicted in Figure 3, this weight is the maximum value among the pixel and its neighbour in the left (L), upper (U) and upper-right (UR) directions. The algorithm uses these three neighbour values to detect edges in horizontal, vertical and diagonal directions. The obtained edge image is sharpened in order to increase the contrast between the background and detected edges; it makes it easy to extract text regions.

UL	U	UR
L	E	R
BL	B	BR

Fig. 3: *Weight for Pixel (x, y).*

The computation of edge image E, algorithm as said in [7] is as follows:

Step 1: Assign left, upper, upper right to 0.

Step 2: For all the pixels in gray image G(x, y) do

Left = (G(x, y)-G(x-1, y))

Upper = (G(x, y)-G(x, y-1))

Upper right = (G(x, y)-G(x+1, y-1))

E(x, y) = max(left, upper, upper right)

Step 3: Sharpen the image E by convolving it with sharpening filter.

Step 4: W(x, y) = Max (L,U,UR).

Localization

The candidate text regions are analysed by horizontal and vertical projection profiles. The

sharpened edge image is considered as the input intensity image for computing the projection profiles, with white candidate text regions against a black background. The vertical projection profile shows the sum of pixels present in each column of the intensity or the sharpened image. In the same way the horizontal projection profile indications the sum of pixels present in each row of the sharpened image. These projection profiles are basically histograms where each bin is a count of the total number of pixels present in each row or column. The vertical and horizontal projection profiles for the sharpened edge image are segmented based on adaptive threshold values, the vertical and horizontal projections Ty and Tx, are calculated, respectively. Regions that fall between the threshold limits are known as candidates for text. The value of threshold Ty is taken into considerations to remove non text regions such as doors, window edges etc. That has a strong vertical orientation. Similarly, the threshold value Tx is selected to eliminate regions which might be non-text or long edges in the horizontal orientation.

$$Tx = \frac{Mean\ (Horizontal\ projection\ profile)}{20}$$

$$Ty = Mean\ (Vertical\ projection\ profile)$$

$$+\frac{Max\ (Vertical\ projection\ profile)}{10}$$

Enhancement and Gap Filling

It is one of the step of connected component method, where the geometric ratio between the width and the height of the text characters is considered to eliminate probable non-text regions. This ratio value will be defined after experimenting on different kinds of images to obtain an average value for calculation. In this step, candidate text regions with the minor to major axis ratio less than ten (10) are considered as for additional processing. Later a gap image is created which will be used as a reference to refine the location of the detected text regions. Gap Filling is defined as, if a pixel in the binary edge image created is surrounded by black (background) pixels in the vertical, horizontal and diagonal directions, this pixel is also substituted with the background value.

PERFORMANCE ANALYSIS

Here the performance of this method is analysed based on the calculation of precision and recall rates from the extracted image.

- ✓ Precision rate is defined as the ratio of correctly detected words to the sum of correctly detected words and false positive. *False positive* are those regions in the image, which are actually not characters of the text but have been detected by the algorithm as text regions.

$$Precision\ Rate = \frac{\text{correctly detected words}}{\text{correctly detected words } + \text{ false positive}} * 100\%$$

- ✓ Recall rate is defined as the ratio of correctly detected words to the sum of correctly detected words and false negative. *False negative (Error Rate):* are those regions in the image, which are actually text characters, but have not been detected by the algorithm.

$$Recall\ Rate = \frac{\text{correctly detected words}}{\text{correctly detected words } + \text{ false negative}} * 100\%$$

RESULTS AND DISCUSSION

Here the 10 images of the dataset taken and the precision and recall rates of those images are calculated for measuring the performance of the method, and the corresponding results are tabulated and results are analysed as shown below section.

Experiments and Analysis

This algorithm has been tested over 10 text images of different types in which text has different font size, color, orientation, alignments. These images are analysed to demonstrate the performance of the proposed algorithm. Performance is verified with the oriented text in horizontal and vertical direction with different languages (English, Korean and Mixed).

Various metrics have been evaluated from the tested results. Here the images were captured either by the use of a high-resolution digital camera or a low-resolution mobile phone camera, and results are shown in the Figure 4. Average precision and recall rate of the method is 65.06 and 89.25% obtained, respectively shown in Figure 5.

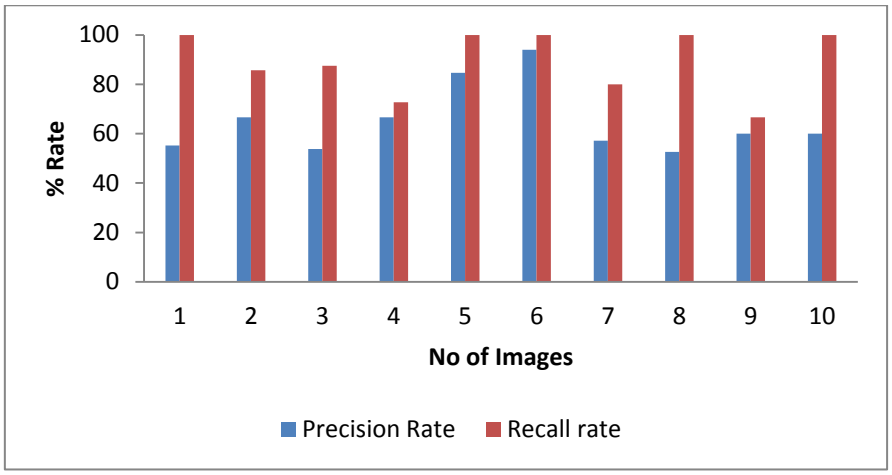

Fig. 4: *Precision and Recall Rates of All Images.*

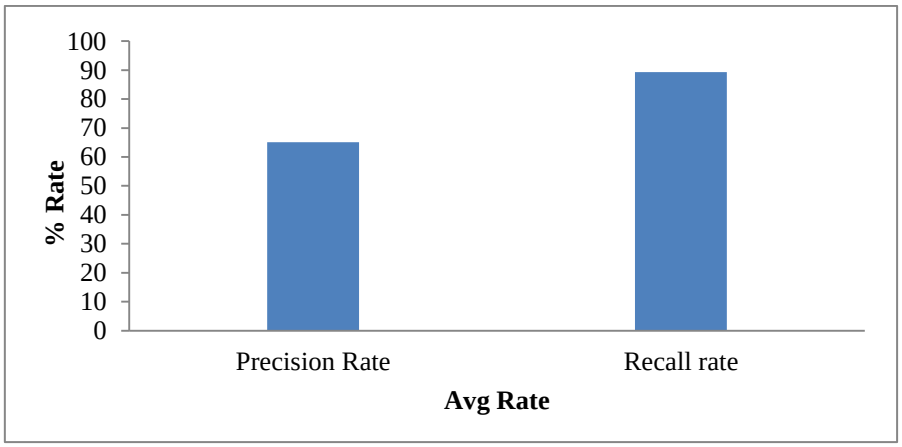

Fig. 5: *Average Precision and Recall Rate.*

CONCLUSION

Hence in this paper the images from scene text images which having different orientations, sizes are taken and the performance of the algorithm is analysed by considering the precision and recall rates of the system. And here overall results for this method for the images are discussed by generating a graph.

REFERENCES

1. Jung K., Kim K.I., Jain A.K. Text Information Extraction in Images and Video: A Survey. *Patt. Recogn.* 2004; 37(5): 977–997p.
2. Lienhart R., Effelsberg W. Automatic Text Segmentation and Text Recognition for Video Indexing. *Multimedia Syst.* 2000; 8: 69–81p.
3. C.M. Lee, Kankanhalli A. Automatic Extraction of Characters in Complex Images, *Int. J. Pattern Recogn.* 1995; 9(1): 67–*82p.*
4. *Jain A. K., Y u B. Automatic Text Location in Image*s and Video Frames. *In Proc. of International Conference of Pattern Recognition (ICPR),* Brisbane, 1998, 1497–1499p.
5. Garcia C., Apostolidis X. T EXT Detection and Segmentation in Complex Color Images. *In Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP2000),* Istanbul, 2000; 4: 2326–2330p.
6. Cai M., Song J., Lyu *M. R. A New Approach for Video Text Detection. In Proc. of International Conference on Image Processing, Roches*ter, New York, USA, 2002; 117–120p.
7. Julinda Gallavata, Ralph Ewerth, Bernd Friesleben, A Robust Algorithm for Text Detection in Images, *Proceedings of 3 International Symposium on Image and Signal Processing and Analysis*, 2003.