IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-6, June 2018 E-ISSN: 2347-2693

Scene Text Extraction using Stroke Width Transform

K. Esther Amulya [1*], P. Sanoop Kumar [2]

[1, 2] Dept. of CSE, Gayatri Vidya Parishad College of Engineering, Visakhapatnam, India

**Corresponding Author: estheramulya@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 12/Jun/2018, Published: 30/Jun/2018

***Abstract*—** The presence of textual components in images is of specific interest which can be extracted using several extraction methods. These components can be helpful for many applications like assisting visually impaired, translator tourists, and robotic navigation in urban areas. The text extraction methods can be classified into three categories: region based, texture based and hybrid method. Extraction based on a region can be further divided into connected component based and edge based method. In spite of numerous scene text detection methods available, 'text extraction' remains unsuccessful. Many issues like different fonts, size, colors, and background noise due to the presence of trees, bricks which are similar to text like objects make text detection difficult. In this paper, the scene text extraction is performed by detecting the edges using canny edge detection algorithm. Then stroke width transform is applied on an edge image with a small yet effective modification in second pass followed by connected component labelling algorithm. The labelled components are then clustered based on the number of pixels available in a particular label. And finally the extracted text is recognized using Google's open source optical character recognition (OCR) engine 'Tesseract'.

***Keywords*—** Text extraction, stoke width, connected component, textual components.

I. INTRODUCTION

We come across a lot of images uploaded online through social media or by digital marketing services. These images contain words or characters. Signboards, banners, store names etc. play an important role and the information these images carry can be advantageous. Consequently, text detection in natural scenes has attracted considerable attention in the computer vision and image understanding community. The textual components can be helpful for many applications like assisting visually impaired, translator for tourists that would translate the text into desired language, scene understanding and robotic navigation in urban areas. The text present in these images is of different format which makes it difficult for the existing OCR engines to recognize it. Currently, available OCR algorithms have been developed to deal with document images in which the text pixels are correctly separated from the background pixels. OCR algorithm expects the input to be black and white image and relatively clean and well-structured [3]. The images have to be refined in a way that makes it easier for the current OCR engines to recognize it. In spite of existence of a lot of text extraction algorithms, it still remains as a challenge due to the presence of different font, size, colours and background noise due to the presence of trees, bricks which are similar to text like objects. There are highly confounding characteristics, such as non-uniform illumination, strong exposure, low contrast, blurring, low resolution, and occlusion, which pose hard challenges for the text detection task [9]**.** All the existing text extraction methods can be classified into three categories namely; region based, texture based and hybrid method [5]. The region based approach attempts to use similarity criterion based on text such as colour, size, stroke width and gradient information to gather pixels. Texture based approaches utilize the distinct textural properties of the text regions to extract candidates sub-windows and finally merging these sub-windows the output is formed. Hybrid approach takes advantage of both region-based approaches which can closely cover text regions and textured-based approaches which can estimate coarse text location in scenes. The region based extraction can be further divided into connected component based and edge based method. The connected component based method like stroke width transform extracts character candidates and group them into word or text lines. Since the focus of these extraction methods is based on connected components rather than texture, factors like illumination, exposure, blurring cannot affect the text detection.

This paper proposes scene text extraction performed by detecting the edges using canny edge detection algorithm. Then for each pixel of edge image stroke width is calculated using stroke width transform with a small yet effective modification in second pass. After generating the stroke width image connected component labelling algorithm is applied. The labelled components are then clustered based on the number of pixels available in a particular label. And finally the extracted text is recognized using Google's open

source optical character recognition (OCR) engine 'Tesseract'.

The goal of the Stroke Width Transform is to calculate the stroke width for each pixel. Since it transforms the image data from containing colour values per pixel to containing the most likely stroke width it is called as Stroke Width Transform [1] i.e. it replaces the pixel values with the stroke width values. Stroke can be defined as a continuous band of nearly constant width. The pixels that have similar stroke width can be clustered together to form bigger components that makes a single character. These components can be labelled and the non- textual elements are rejected based on some criteria like calculation of aspect ratio using width and height of connected component, width variation by computing mean and variance etc.

The rest of the paper is organized as follows: Section II contains the related work of other text extraction methods; Section III contains proposed system architecture and methodology; Section IV contains the results and discussions.

II. RELATED WORK

There are two types of region based extraction method which is connected component based and edge based method. The connected component based method checks the difference between the text and the background to extract connected regions and then uses heuristic rules such as aspect ratio, size and geometric functions to filter the non-text connected regions. This has become the mainstream text detection method as more attention has been paid to these methods. Among this approach, Stroke Width Transform (SWT) and maximally stable extremal regions (MSER) are the most widely used basic detection algorithms because of their efficiency and stability.

The work done by Canedo *et al.* in [6] detected text using frequency information of the Discrete Cosine Transform coefficients, binarized it using clustering-based algorithm and then recognize it by use of an optical character recognition algorithm. The method proposed by them detects text regions in image using frequency information of the luminance DCT 8x8 bock of a JPEG image. After which text energy is calculated and their mean is found out. A Gaussian function centred on the vertical middle of the image is imposed in a way that the text block candidate around of the vertical middle defines more probable text that is of interest. Then the text is binarized and recognized.

Sarwar Khan *et al.* [7] proposed a text recognition method based on Support vector Machine (SVM), KNN and maximally stable extremal regions (MSER). They used some feature for panel and to train SVM. MSER is used to segment each potential character present in the image. To filter out the non-character elements height, width, size, aspect ratio and stroke width are used. The classifier takes 32x32 pixel bitmap and classifies the characters of different languages. Similarly KNN a non-parametric classification method is used for character classification and text recognition.

In [8] the input image is first transformed into a binary image and edge detection is applied. Instead of performing a simple thresholding method, Maximally Stable Extremal Regions (MSER) is detected. These regions contain the text components and are appointed as white pixels. Since the resulting binary image does not reveal the exact boundaries of text, MSER binary image is enhanced by performing a thresholding operation on each connected component. Edges are then detected and fed into a stroke width detector where strokes, stroke widths, and connected components are found and filtered. Furthermore, text lines are formed prior to text extraction in order to cut down more non-text pixels and increase the accuracy. Similarly work done in [10] proposes a coherent framework for addressing automated text recognition.

III. METHODOLOGY

The proposed system is developed to take a JPEG image as input. The image is processed in order to extract text using stroke width transform followed by connected component labelling. The text is then recognized using Tesseract engine. Fig.1 depicts the architecture of proposed system.

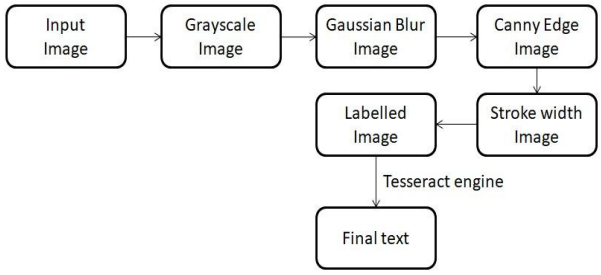


Figure 1. *Block diagram*

A. Text extraction

The proposed system is developed to take a JPEG image as an input. The image is processed in order to extract the text and recognize it. The text extraction is divided into 4 stages:

1) Canny Edge detection.
2) Stroke width image generation.
3) Connected component labelled image.
4) Reject non-text elements.

1) Canny Edge detection.

Canny edge detection algorithm is used to detect the edges in the image. Firstly, the image is smoothed using Gaussian filter which reduces the effect of noise in the image giving a blurred image. This blurred image is used to calculate the gradient direction (G_x and G_y) using Sobel operators and magnitude is calculated using the gradients ($G=\sqrt{G_x^2 + G_y^2}$) [2]. In order to find the sure shot edges, find the local maxima (higher magnitude). Then select lower and higher threshold values and consider the values which fall between these thresholds.

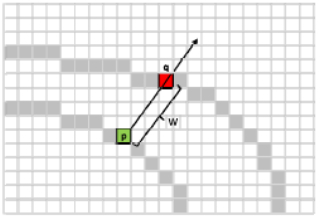
(a) (b)

Figure 2. *(a) Input JPEG image (b) Edge image after applying canny edge detection algorithm*

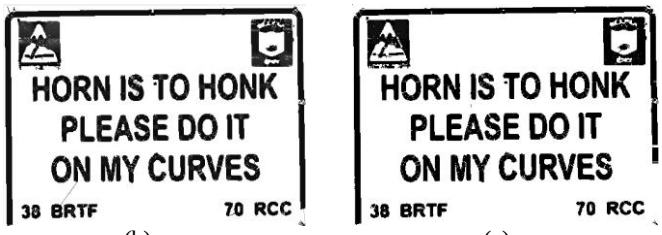
2) Stroke width image generation.

The stroke width transform is a local image operator which calculates the most likely stroke width for pixel. A stroke can be defined to be a contiguous part of an image that forms a band of nearly constant width [1]. After computing the edges in the image the initial value for each element of SWT is set to infinity (∞). For every pixel p, if it lies in the stroke boundary consider its gradient dp. Follow the ray along the direction of the gradient until another edge pixel q is found as shown in Fig. 3(a). This second pixel should be roughly opposite to the first pixel p. Each pixel along the ray is assigned with the recorded width of the stroke i.e. distance between p and q. Pass along the previously non-discarded rays and compute the median stroke-widths of all pixels. Some complex situations like corner exist for which stroke widths will not be true in the first pass. For this reason median stroke width values is computed. All the pixel values that are above median are made equal to median.

The most important modification to the classical stroke width transform discussed above is that all the recorded stroke-widths are grouped. Each cluster has a minimum and maximum stroke width value and pixels whose values are nearer falls into one cluster. The cluster with maximum pixel count is considered as the average width of the text present in the image.



(a)



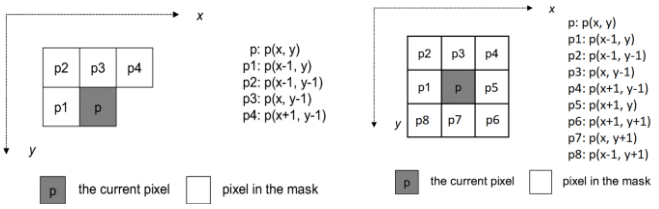
(b) (c)

Figure 3. *(a) p is a pixel on the boundary of stroke. Searching in the direction of gradient at p, leads to finding q, the corresponding pixel on the other side of the stroke [1]. (b) Stroke width image without modification and (c) stroke width image with modification in pass 2. It can be seen a lot of lines have been removed from (b) which in turn gives a clearer text as seen in (c).*

3) Connected component labelling.

To form the connected components the SWT image is labelled using Label-Equivalence-Based Two-Scan Labelling Algorithm [4]. First, scan through the stroke width image from left to right and top to bottom. If the current pixel is not a background pixel then check for its neighbours using the mask in Fig. 4(a). If none of the neighbours are found then assign the current pixel with a unique label. Otherwise, find the neighbour with minimum label value and assign it to the current pixel. Store the equivalences between neighbouring labels i.e. wherever conflicts are present. In the second pass again scan through all the elements of the image. Now check the neighbours using the mask shown in Fig. 4 (b). Get the minimum label value and assign it to current pixel.

Our approach differs from the one discussed in [4] as a refinement pass is performed. This pass is performed in order to remove the background noise and obtain a clear separate text component. In this refinement pass the 8- neighbours of a current pixel p as shown in Fig. 4(b) are checked. If more than three neighbouring pixels are non-background pixels then find the minimum label value from the neighbours and assign it to the current pixel p. In a case where five or more than five neighbours are background pixels then make the current pixel as background pixel.



(a) (b)

(c) (d)

Figure 4. *(a) Mask for labelling [4] (b) Mask for labelling using 8-neighbors (c) Labelled Image after pass 2 (d)Labelled Image after refinement pass. When (c) and (d) are compared a lot of black spots present in(c) appear to be filled in (d).*

4) Rejection of non-text elements

While forming connected components there can be a lot of components which are not the part of text. These components can be rejected using set of some rules.

- Calculate the width and height of the connected component and find its aspect ratio.

  $Aspect\_Ratio = \min\{h(c)\ w(c), w(c)\ h(c)\}$
  Its value must be between 0.1 and 10 [1].

- Compute the mean and variance of each connected component if variance is greater than half of the mean then reject it.

After rejecting the components which does not satisfy the above rules, the remaining components comprise the text. In order to detect texts of different sizes, form clusters based on the number of pixels present in each component with a label. Consider the clusters with maximum pixels and reject the others.

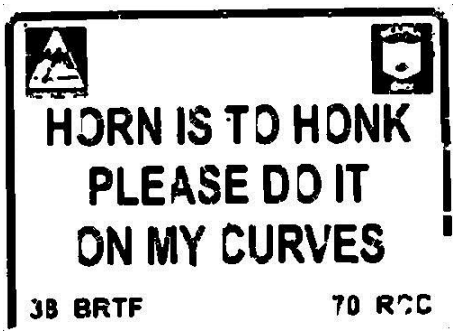


Figure 5. *Final image obtained after applying the set of rules*

B. Text recognition

The of the most accurate optical character recognition engines available is "Tesseract". It works for various operating systems. It was initially developed at Hewlett Packard (HP) labs. As very little work was done by them in later years it was released as an open source engine and since 2006 it has been developed and improved by Google. It supports optical character recognition for TIFF, JPEG, BMP image formats and pdf document format.
To get started with tesseract, download the tess4j folder and add all the required JAR files into the project. Give the connected component image and text will be recognized. The recognized text for Fig. 4 is as show below:

HORN IS TO
HONK
PLEASE DO IT
ON MY CURVES
JB BRTF 70 R'IC

Figure 6. *Text recognized by "Tesseract"*

IV. RESULTS AND DISCUSSION

Initially a JPEG image is take is taken. The edges are detected using canny edge detection algorithm. The stroke width transform is applied to the edge image; this stroke width image is not clean enough for text detection. To get a eliminate background noise connected component labelling is done which is followed by a cluster formation based on number of pixels available in each component. Finally the extracted text was recognized using "Tesseract".

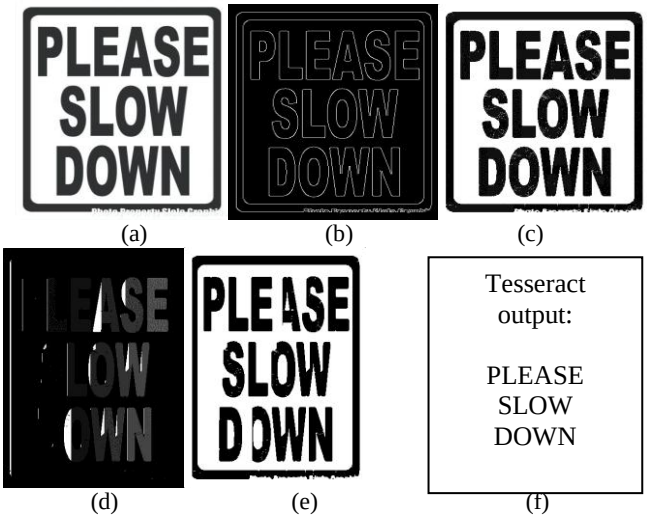


Figure 7. *(a) Input image (b) Canny Edge Image (c) Stroke width Image (d) Labelled Image (e) Final Image (f) Tesseract output*

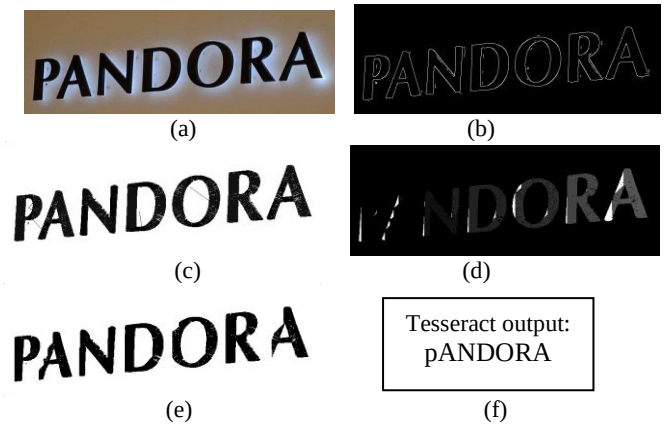


Figure 8. *(a) Input image (b) Canny Edge Image (c) Stroke width Image (d) Labelled Image (e) Final Image (f) Tesseract output*

V. CONCLUSION AND FUTURE SCOPE

Scene text extraction is a recent research area in the field of computer vision. Text extraction is challenging due to different variety of text patterns like font, size, colour and the presence of background noise like trees, bricks etc.

In this paper, text extraction is done using stroke width transform. Connected components are formed from the stroke width image to get separate characters of the text. Some heuristic rules are applied to remove the non-textual elements. Finally, in order to detect text of different size clustering-based method is applied. The proposed system faces challenges when focus is on the text is less and a lot of other background noise is present. Another situation where text detection gets difficult is when the text is in the background and a lot of other objects are present in the foreground. Modification can be done in order to overcome these problems.

REFERENCES

[1] B. Epshtein, E. Ofek, and Y. Wexler, "*Detecting text in natural scenes with stroke width transform,*" IEEE conference on Computer Vision and Pattern Recognition, pp. 2963-2970, June 2010.

[2] John H Canny, "*A Computational approach to Edge Detection,*" IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. PAMI-8, NO. 6, NOVEMBER 1986

[3] Pooja Chavre, Archana Ghotkar, "*Scene Text Extraction using Stroke Width Transform for Tourist Translator on Android Platform,*" 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), pp. 301-306, 2016.

[4] Lifeng He, Yuyan Chao, and Kenji Suzuki,"*Two Efficient Label-Equivalence-Based Connected-Component Labeling Algorithms for 3-D Binary Images,*" IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 20, NO. 8, AUGUST 2011

[5] K. Jung, K. Kim, A. K. Jain, "*Text information extraction in images and video: a survey*", Pattern Recognition, p. 977 – 997,Vol 5. 2004.

[6] Adrian Canedo, Jung H. Kim, ,Soohyung and Yolanda Blanco-Fernández "*English to Spanish Translation of Signboard Images from Mobile Phone Camera,*" IEEE conference, Southeastcon, pp. 356-361, Mar. 2009.

[7] Sarwar Khan and Somying Thainimit, "*Text Detection and Recognition on traffic panel in roadside imagery*," International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES), pp. 1-6, 2017

[8] Najwa-Maria Chidiac, Pascal Damien, Charles Yaacoub, "*A Robust Algorithm for Text Extraction from Images,*" International Conference on Telecommunications and Signal Processing (TSP), pp. 493- 497, 2016.

[9] ZhuoyaoZhong, LianwenJin, Shuangping Huang, "*DeepText: A new approach for text proposal generation and text detection in natural images,*" IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1208-1212, 2017.

[10] Pooja Kumari, Mamta Yadav, "Detection and Recognition for Reading Text in Images", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 3, Issue 5, pp.980-984, May-June.2018

Authors Profile

Mr. P. Sanoop Kumar received B.Tech. degree from Vignan's Institute of Information Technology, Duvvada in 2007 and M.Tech from Andhra University in the year 2011. He is currently working as Assistant Professor in the Department of Computer Science & Engineering at Gayatri Vidya Parishad College of Engineering(Autonomous), Visakhapatnam since 2012. He has 6 years of teaching experience.

Miss. K.Esther Amulya received B.E in Compter Engineering from Pillai's College of Enginnering, Navi Mumbai in 2016. She is currently pursuing M.Tech from Gayatri Vidya Parishad College of Engineering, Visakhapatnam, A.P.

