INTERNATIONAL JOURNAL FOR TRENDS IN ENGINEERING & TECHNOLOGY
VOLUME 3 ISSUE 3 – MARCH 2015 – ISSN: 2349 – 9303

Scene Text Detection of Curved Text Using Gradiant Vector Flow Method

S. Sudhakar Ilango
Dept of Computer Science
Sri Krishna College Of Engineering and Technology
Coimbatore, India
Sudhakar.ilango@gmail.com

Kalaivani L
Dept of Computer Science
Sri Krishna College of Engineering and Technology
Coimbatore, India
Kalaiforevr@gmail.com

Abstract--Text detection and recognition is a hot topic for researchers in the field of image processing and multimedia. Content based Image Retrieval (CBIR) community fills the semantic gap between low-level and high-level features. For text detection and extraction that achieve reasonable accuracy for multi-oriented text and natural scene text (camera images), several methods have been developed. In general most of the methods use classifier and large number of training samples to improve the accuracy in text detection. In general, connected components are used to tackle the multi-orientation problem. The connected component analysis based features with classifier training, work well for achieving better accuracy when the images are highly contrast. However, when the same methods used directly for text detection in video it results in disconnections, loss of shapes etc, because of low contrast and complex background. For such cases, deciding geometrical features of the components and classifier is not that easy. To overcome from this problem the proposed research uses Gradiant Vector Flow and Grouping based Method for Arbitrarily Oriented Scene text Detection method. The GVF of edge pixels in the Sobel edge map of the input frame is explored to identify the dominant edge pixels which represent text components. The method extracts dominant pixel’s edge components corresponding to the Sobel edge map, which is called Text Candidates (TC) of the text lines. Experimental results on different datasets including text data that is oriented arbitrary, non-horizontal text data also horizontal text data, Hua’s data and ICDAR-03 data (Camera images) show that the proposed method outperforms existing methods.

Index Terms—Connected component (CC)-based approach, CC clustering, machine learning classifier, Gradiant vector Flow method, Sobel Edge Map, non-text filtering, scene text detection.

I. INTRODUCTION

Image Processing is a technique to enhance raw images received from cameras/sensors captured in normal day-to-day life for various applications. Techniques have been developed in Image Processing during the last four to five decades. Most of the techniques are developed for enhancing images obtained from military reconnaissance flights, and many other sources. Due to easy availability of graphics software, large size memory devices, powerful personnel computers etc., image processing systems are becoming popular. In computer vision community, Text detection and recognition in camera captured images have been considered as very important problems. The reason for this kind of problems is that the text information recognized by machines and variety of applications can be used for various needs. Some examples are tourists can easily translate information by means of retrieval systems in indoor and outdoor environments, people who are visually impaired, robot that can automatically navigate etc.,. Even though there are lot of research activities in this field, scene text detection is still remained as a critical problem. The reason behind this is, text images in the scene normally suffer from photometric degradations as well as geometrical distortions so that many algorithms faced the accuracy and/or speed (complexity) issues.

A. *Text Detection in Videos and Natural Scenes*

In natural scene images, texts provide highly useful information and can provide a key for understanding the content of image. The increasing growth of video data creates a need for efficient retrieving systems and content-based browsing. Text that are in various form is embedded frequently into images to provide important information about the scene like people names, identification of documents such as geographical maps or engineering graphics, titles of any image, date of an event in natural scene images and news video sequences, road-sign interpretation etc. Hence, text needs to be detected for semantic understanding and image indexation. Perform the detection of text in a natural scene image or a video frame can be achieved in few major steps such as, feature extraction of the image, classification of the text and image pre-processing.

Candidate regions are identified by applying HTE Pre-processing techniques. By means of using the optimal edge detector, Edge information is extracted. This helps in finding connected components precisely by removing useless edge pixels. After finding components that are connected, filtering is applied character block to screen those connected components that do not contain texts.

Text extraction and verification is done by means of feature extraction. There exists, two sets of features; the first feature set uses the discrete wavelet transform (DWT) to extract the features of characters for texture analysis along with its description. The next feature is a new one that is local energy based shape histogram 'LESH'. In order to provide training samples to the feature extraction, a set of training samples that contains 2000 text images and 650 non text images data gathered from different images and video streams are collected.

By means of using nearest mean classifier the classification is done; however, adaptive adaboost algorithm is used for classification purpose to enhance results.

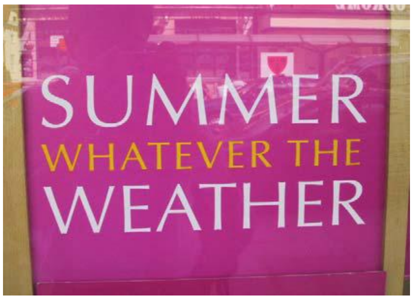

Fig.1.a. Input image

B. *Problem of Text Extraction*

The problem of Text Information Extraction (TIE) needs to be defined more precisely before proceeding further. A TIE system receives an input in the form of a still image or a sequence of images. The images can be in compressed or Un-compressed format, color or gray scale, and the text in the images may or may not move. The TIE problem can be divided into the following sub-problems: detection of text, localization of text, Tracking the text, extraction, enhancement and recognition (OCR).

Text detection, localization, and extraction are often used interchangeably in the literature. However, in this paper, these terms are differentiated. Text detection refers to the determination of the presence of text in a given frame. Text localization is the location of text in the image. Around the text, bounding boxes are generated. Text tracking is the process of reducing the processing time for text localization and to maintain the integrity of position across frames at adjacent position. Even though the location of text in an image can be indicated by bounding boxes, to facilitate its recognition, the text still needs to be segmented from the background. This means that before the extracted text image is fed into an OCR engine, it has to be converted to a binary image and enhanced. Text extraction is the process where the text components are segmented from the background.

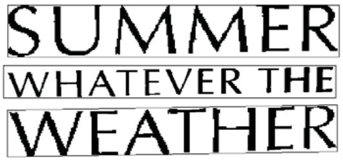

Fig.1.b. Illustration of Ground truth

II. PROPOSED METHODOLOGY

The proposed research presents a new method that extracts text lines of any orientations based on neighbor component grouping and Gradient Vector Flow (GVF). The Sobel edge map's GVF of edge pixel of the input frame is explored to identify the dominant edge pixels which represent text components. The edge components corresponding to dominant pixels in the Sobel edge map are extracted by this method, which is called Text Candidates (TC) of the text lines.Two grouping schemes are proposed. The first finds nearest neighbors based on geometrical properties of TC to group broken segments and neighboring characters which results in patches of word. To eliminate false positives the end and junction points of skeleton of the word patches are considered, that results in Candidate Text Components (CTC). The second is based on the direction and the size of the CTC for restoring missing CTC and to extract neighboring CTC, which enables arbitrarily-oriented text line detection in video frame. Results of experiments on different datasets including Hua's data and ICDAR-03 data, non-horizontal, arbitrarily oriented text data and horizontal text data, (Camera images) show that the proposed method outperforms existing methods in terms of performance.

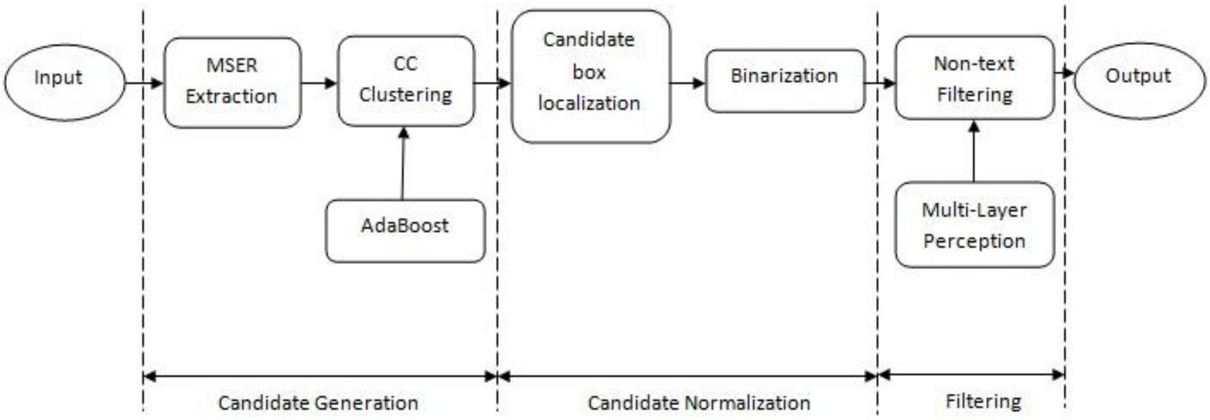

Fig.2. Block Diagram of Proposed Methodology

A. Image Capturing

The input images with text and non-text component are initially accrued by the mobile camera. Resolution of the image is fixed to be 640 × 480 so that any basic camera can be used for the purpose of taking image. Also, it takes about 120 ms when three channels are used.

B. *Pre Processing*

If there is any noise corrupted in the image, they are initially removed by noise removal of median filter. This filtering technique is efficient and simple for noise such as impulse or salt and pepper in the image.

C. Candidate Generation

For the candidate generation, Connected Components (CCs) are extracted in images and partition the extracted CCs into clusters, where the clustering algorithm is based on an adjacency relation classifier. The approaches discuss (i) to build samples for training, (ii) training the classifier and (iii) to use that classifier in the CC clustering method.

CC Extraction using MSER

MSER algorithm can be considered as a process to find most of the text components and also to find local Binarization results that are stable over a range of thresholds. The MSER algorithm yields CCs that are either darker or brighter than their surroundings

Building Training Sets

The classifier is based on pair-wise relations between CCs; let us first consider cases that can happen for a CC pair
(ci, cj) ∈ C × c(i!=j):
1) ci ∈ T , cj ∈ T , ci ~ cj
2) ci ∈ T , cj ∈ T , ci !~ cj , t (ci) = t (cj)
3) ci ∈ T , cj ∈ T , ci ~ cj , t (ci) != t (cj)
4) ci ∈ T , cj ∈ N
5) ci ∈ N, cj ∈ N.
Specifically, sets of CCs are obtained by applying the MSER algorithm to a training set released. Then, for every pair (ci, cj) ∈ C × c(i! = j), category among 5 cases are identified. By means of gathering samples corresponding to the case (1) and a negative set by gathering samples corresponding to the case (3) or (4) a positive set is built. Samples from other cases were discarded.

D. CC Clustering

With the collected samples, an AdaBoost classifier is trained that tells us whether (ci, cj) ∈ C×C (i! = j) is adjacent or not. Let us define some local properties of CCs in order to present the method. Given ci ∈ C, a bounding box of ci is found. Denote its width and height as wi and hi respectively. Given a pair such that (ci, cj) ∈ C×C, horizontal overlap, vertical overlap and the horizontal distance between two boxes are denoted as dij , hoi j, and voij respectively.
The function of AdaBoost algorithm is,
$\Phi: C \times C \rightarrow \mathbb{R}$
In binary decisions the following function is used:
$\Phi\ (ci,\ cj) > \tau 1 \Longleftrightarrow ci \sim cj$
With a threshold $\tau 1$, all adjacent pairs are found by evaluating that function for all possible pairs in *C*, Given $\varphi\ (\cdot,\ \cdot)$ and $\tau 1$. *C* is partitioned into a set of clusters based on these adjacency relations,
$W = \{wk\}$
Where $wk \subset C$. Normally, $ci, cj \in wk$ (i.e., *ci* and *cj* are in the same cluster) means that there exists $\{ei\}^{m}_{i=1} \subset C$ such that
$ci \sim e1 \sim e2 \sim \cdot \cdot \cdot em \sim c\,j$
By means of using the union-find algorithm, W can be built. After clustering, clusters having only one CC are discarded.

E. Candidate normalization

After CC clustering, there will be a cluster set. Corresponding regions for the reliable text/non-text classification in this section are normalized.

Geometric Normalization

First localize its corresponding region given wk ∈ W. Approximate the shape of text boxes with parallelograms whose left and right sides are parallel to *y*-axis, if although the text boxes can experience perspective distortions. This approximation alleviates difficulties in estimating text boxes having a high degree of freedom (DOF): Find a skew and four boundary supporting points. Build two sets given below, for estimation of a given word candidate *wk's* skew.
$Tk = \{t(ci)\ |ci \in wk\}$
$Bk = \{b(ci)\ |ci \in wk\}$
Where *t(ci)* and *b(ci)* are the top-centre point and the bottom-centre point of a bounding box of *ci*, respectively.

Binarization

Build binary images, given geometrically normalized images. In many cases, MSER results can be considered as Binarization results. However, perform the Binarization separately by estimating background colors and text. The reason for this is that (i) some character components and/or yield noisy regions (mainly due to the blur) may be missed by the the MSER results and (ii) store the point information of all CCs for the MSER-based Binarization. The average colors of CCs are considered as the text color:

$$\frac{\sum_{c_i \in w_k} s_i \mu_i}{\sum_{c_i \in w_k} s_i} \in \Re^3$$

Consider the average color of an entire block as the background color. Then, obtain a binary value of each pixel by comparing the distances to the estimated text color and the estimated background color.

F. Filtering

In order to get final results develop a text/non-text classifier that rejects non-text blocks among normalized images.

Multilayer Perceptron Learning

For the training, normalized images are needed. For this goal, apply algorithm presented to the training images. Then, text and non-text are manually classified. Some images showing poor Binarization results are discarded, and collected 863 non-text block images and 676 text block images. By applying the same procedure to images that do not contain any text, more negative samples are needed for the reliable rejection of non-text components and collected more negative samples. Then divide the text/non-text images into squares and train a multi-layer perceptron for the classification of square patches.

Proposed filtering: Filtering text from non-text of text

In order to identify dominant text pixel using Sobel edge map of the input image for arbitrary text detection in the image, explore GVF.

GVF for Dominant Text Pixel Selection

The Gradient Vector Flow (GVF) is a vector that minimizes the energy functional as defined in following equation
Where h(x, y) = (a(x, y), b(x, y)) is the GVF field and is the edge map of the input image.

Size:

$$\frac{medianLength(g)}{3} < length(c) < medianLength(g) \times 3$$

Text Candidates Selection

For text candidate selection, the result of dominant pixel selection is used. The method extracts edge components from the Sobel edge map corresponding to dominant pixels for each dominant pixel. These extracted edge components are called text candidates. Then the extracted text candidates are used in the next section to restore complete text information with Sobel edge map.

Grouping of Candidate Text Components

For each text candidate the method finds its perimeter and it allows five iterations for the perimeter to grow, pixel by pixel, in the direction of the text line in the Sobel edge map of the input frame to group neighboring text candidates. Contour of the text candidates is defined as the perimeter. Minor axis for the perimeter of the text candidates is computed by the method and it considers length of the minor axis as radius to expand the perimeter. The method traverses the expanded perimeter to find the text pixel (white pixel) of the neighboring text candidate in the text line at every iteration. This step will merge segments of character components and neighbor characters to form a word.

Text candidates which have close proximity within five iterations of the perimeter will be merged by this process. By studying the space between the text candidates, the five is determined empirically. The tolerance of five pixels is acceptable as it is lower than the space between the characters. As a result, there will be two groups of text candidates, that is the current group and the neighbor group. Then the following properties are verified based on the size and angle of the text candidate groups before merging them. Generally, the major axes of the character component's length will be almost the same lengths and the angle difference between the character components has almost the same angle. However, fix $\theta_{\min 2}$ as 10^0 because in case of arbitrarily oriented text,

$$\mathcal{E} = \iint \mu\left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right) + |\nabla f|^2 |g - \nabla f^2| \, dxdy$$

According to nature of text line orientation, there will be slight different orientations for each character. Fix the 5^o, to take care of little orientation variation.

III. RESULT

Experimental results on different datasets including text data that has various orientation, text data that is not horizontal, Hua's data and ICDAR-03 data (Camera images) and horizontal text data show that the proposed method outperforms existing methods in terms of F-measure, precision and recall. This system has much advancement over the existing system and the following observations are confirmed.

- Non-text information in complex background will be removed.
- Text pixels are not missed.
- Better accuracy for text detection.
- Text detection in video can also be achieved.

IV. CONCLUSION

In the present work, Gradient Vector flow is proposed for detecting the curved text detection. For the first time text detection in Image by selecting dominant text pixels and text candidates with the help of the Sobel edge map is done with the Gradiant Vector Flow information. Non-text information in complex background of Images can be removed by the dominant text pixel selection. To be precise, two classifiers are developed: one classifier was designed to generate candidates and the other classifier was for the filtering of non-text candidates. A novel method to exploit multi-channel information is presented. The conducted experiments on datasets showed that the proposed method yielded the state-of-the-art performance in both new and traditional evaluation protocols.

REFERENCES

[1]. J. Zang and R. Kasturi, "Extraction of Text Objects in Video Documents: Recent Progress", In Proc. of DAS 2008, pp 5-17.

[2]. K. Jung, K.I. Kim and A.K. Jain, “Text information extraction in images and video: a survey”, Pattern Recognition, 2004, pp. 977-997.

[3]. Crandall and R. Kasturi, “Robust Detection of Stylized Text Events in Digital Video”, In Proc. of ICDAR 2001, pp 865-869.

[4]. K. L Kim, K. Jung and J. H. Kim. “Texture-Based Approach for Text Detection in Images using Support Vector Machines and Continuous Adaptive Mean Shift Algorithm”. IEEE Transaction on PAMI, 2003, pp 1631-1639.

[5]. Wang, C. W. Ngo and T. C. Pong, “Structuring low quality videotaped lectures for cross-reference browsing by video text analysis”, Pattern Recognition, 2008, pp 3257-3269.

[6]. U. Bhattacharya, S. K Parui and S. Mondal, “Devanagari and Bangla Text Extraction from Natural Scene Images”, In Proc. of ICDAR 2009, pp 171-175.

[7]. X. Chen, J. Yang, J. Zhang and A. Waibel, “Automatic Detection and Recognition of Signs from Natural Scenes”, IEEE Transactions on Image Processing, 2004, pp 87-99.

[8]. Y. F. Pan, X. Hou and C.L. Liu, “A Hybrid Approach to Detect and Localize Texts in Natural Scene Images”, IEEE Transactions on Image Processing, 2011, pp 800-813.

[9]. Y. F. Pan, X. Hou and C. L. Liu, “Text Localization in Natural Scene Images on Conditional Random Field”, In Proc. of ICDAR 2009, pp 6-10.

[10]. B. Epshtein, E. Ofek and Y. Wexler, “Detecting Text in Natural Scenes with Stroke Width Transform”, In Proc. of CVPR, 2010, pp 2963-2970.