

VIDEO TEXT DETECTION BASED ON FILTERS AND EDGE FEATURES

Palaiahmakote Shivakumara, Trung Quy Phan and Chew Lim Tan
School of Computing, National University of Singapore
{shiva, phanquyt, tancl}@comp.nus.edu.sg

ABSTRACT

Text detection plays a vital role in retrieving and browsing video data efficiently and accurately. In this paper, we propose a method for detecting both graphics and scene text in video images by proposing initial text block identification, text portion segmentation and new edge features for false positive elimination. The heuristic rules based on filters and edge analysis are formed to identify the initial text block and to segment the complete text portion from the image. The new edge features such as straightness and cursiveness are explored to eliminate false positives. To evaluate the performance of the proposed method, we introduce misdetection rate and processing time in addition to detection rate and false positive rate. The experimental results show that the proposed method outperforms existing methods in terms of the above metrics.

Index Terms— Initial text block detection, Text portion segmentation, Canny edge profiles, Edge features, Text detection.

1. INTRODUCTION

Nowadays, the advancement of technology and decreasing price of the media devices leads to videos capturing of many day to day activities. As a result, hundreds of thousands of hours of archival videos are being stored and shared. This situation demands for tools that allow efficient and accurate browsing and retrieving of video data [1-9]. Two types of text in video are: *caption/graphics/artificial* text which is artificially superimposed on the video at the time of editing, and *scene* text which naturally occurs in the field of view of the camera during video capture. The state of the art [1-3] on text detection in video images shows that detection of both graphic and scene text in the video images under different situations is still challenging and an emerging problem in the field of information retrieval due to complex background, low resolution and low contrast. The existing methods such as connected-component based methods [4], texture based methods [5, 6], uniform color based methods [7], gradient based methods [8], and edge based methods [9] work well for the specific data under controlled environment. However, if the dataset includes the images with complex background, low resolution and poor contrast, the performance of existing methods degrades severely.

2. PROPOSED METHODOLOGY

In this work, we consider only horizontal text lines for text detection in video images. The portion of the text is segmented using the method proposed in [10]. For segmented text portion, we propose a method for text detection. The interesting part in this work is the use of canny edge profiles for fixing bounding boxes for the text lines in the image and elimination of false positives by

introducing new edge features without any complex mathematics. This is because we observe that the Canny detector produces edges even for low contrast image but at the cost of false positives. To eliminate false positives, we propose new edge features such as straightness and cursiveness. It is also noted from the paper [11] that false positive elimination is challenging and interesting. Hence, in this paper, we focus on an efficient method for fixing bounding boxes for text lines in image and new way for false positive elimination. Segmenting text portions from the image by proposing filters and edge analysis is an added advantage for the proposed method comparing to existing methods.

2.1. Initial text block detection

We first divide a given image of size 256×256 into 16 equal sized blocks as shown in Figures 1 and 2. We then apply several operations on the image blocks to derive heuristic rules for segmenting candidate text block selection. This is illustrated in Figure 3 where (a)-(f) show the various operations leading to $S_{AF}(x,y)$ and $C_{Diff}(x,y)$, where S_{AF} is the Sobel edge block for the output of arithmetic filter [12] and C_{Diff} is the canny edge block for the difference block (*Diff*) which is subtraction of the arithmetic filter from the median filter of the block. Let NS_{AF} and NC_{Diff} be the number of edges in $S_{AF}(x,y)$ and $C_{Diff}(x,y)$, respectively. We therefore have the following rule:

$$R_1 = \begin{cases} \text{Text Block}, & \text{if } NS_{AF} > NC_{Diff} \\ \text{NonText Block}, & \text{Otherwise} \end{cases} \quad (1).$$

Figure 3(g)-(l) show the various operations leading to $W_{MF}(x,y)$ for the median filter block $MF(x,y)$ and $W_{Diff}(x,y)$ for the *Diff* block. The number of strong edges with respect to *MF* is computed by subtracting the number of edges in $W_{MF}(x,y)$ from the number of edges in $C_{MF}(x,y)$, where $W_{MF}(x,y)$ is the subtraction of the Sobel edge block from the Canny edge block on *MF* block and $C_{MF}(x,y)$ is the canny edge block that corresponds to *MF* block. Similarly, the number of strong edges with respect to *Diff* block using $W_{Diff}(x,y)$ is obtained. Let NST_{MF} and NST_{Diff} be the numbers of strong edges that correspond to *MF*(x,y) and *Diff*(x,y), respectively. Thus another rule (R_2) is defined as

$$R_2 = \begin{cases} \text{Text Block}, & \text{if } NST_{MF} > NST_{Diff} \\ \text{NonText Block}, & \text{Otherwise} \end{cases} \quad (2).$$

It is noticed from Figure 4 that R_1 classifies (positive difference values) almost all text blocks 14, 15 and 16 in Figure 2 as text blocks except block 13. R_2 also classifies the blocks 13, 14, 15 and 16 as text blocks but it misclassifies other blocks as text blocks. Hence, we use both R_1 and R_2 to identify the initial text block out of 16 blocks. Let D_1 be the set of difference values between the NS_{AF} and NC_{Diff} and HD be the highest value in D_1 . From the D_1 , the HD is chosen and if a block corresponding to HD in D_2 ($NST_{MF} - NST_{Diff}$) gives a positive difference value then it is considered as an initial text block for segmenting the complete text portion from

the image. This is illustrated in Figure 4 where it is noticed that the 15th block satisfies both criteria. If a block does not satisfy this criterion then the method searches the next highest difference in D_j .



Figure 1. Video image



Figure 2. 16 Blocks

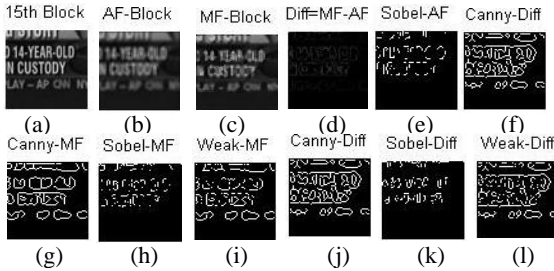


Figure 3. Initial text block selection

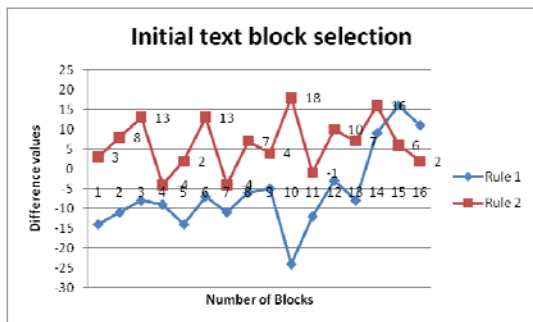


Figure 4. Illustration for initial block detection

2.2. Text portion separation

The method uses R_2 ($NST_{MF} > NST_{Diff}$) to grow initial text block boundary. But both rule R_1 and R_2 are used for stopping the boundary growing. First, the boundary grows towards right and then left, as shown in Figure 5. Then the boundary grows upward in a similar way as shown in Figure 6 and the final boundary is also shown in Figure 6. If all three conditions fail then we consider the whole image as the segmented portion. Such example can be seen in Figure 11. The procedure works for most of the images collected in our dataset. More details for segmenting text portion can be found in [10].



Figure 5. Block growing towards right and left direction



Figure 6. Boundary growing upward direction and final boundary of the image



Figure 7. Text detection

2.3. Text block detection using edge profiles

The method obtains a Canny edge map for the segmented text portion image. The small edges containing less than a threshold $t1$ pixels are removed from the edge map as shown in Figure 7(a). The method detects the text blocks by analyzing the projection profiles with the information about the text alignment of the filtered edge map as shown in Figure 7(b). Finally, the true text blocks are extracted by removing false positives as shown in Figure 7(c). The Horizontal Profiles (HP) for the Canny edge map (CM) is defined as $HP = \sum_{y=1}^n \sum_{x=1}^m CM(x, y)$, where n and m are the

dimension of the CM . The method chooses entries in HP whose length is greater than a threshold $t2$ pixels. Let $P1$ and $P2$ be such profiles. Vertical Profiles (VP) are then generated between $P1$ and $P2$ as $VP = \sum_{x=1}^m \sum_{y=P1}^{P2} CM(x, y)$.

If the gap between two adjacent non-zero VPs is greater than a threshold $t3$ pixels, the method adds boundary between the two VPs. $t3$ pixels is the maximum allowed space between two characters in the same word.

2.4. False positive elimination

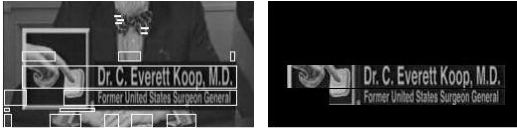
Detection of text blocks using Canny edge profile causes a larger number of false positives. However, due to segmentation of text portions, fewer number of false positives exist. One such example is shown in Figure 8(a). In order to remove false positives, we introduce new edge features namely straightness and cursiveness, apart from the height and width of the text blocks. Let $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$ be the sets of x and y co-ordinates respectively, of the pixels belonging to an edge. The centroid of an edge is (C_x, C_y) defined as $C_x = \frac{1}{n} \sum_{i=1}^n x_i$ and $C_y = \frac{1}{n} \sum_{i=1}^n y_i$, where n

is the number of pixels in the edge. Using this definition of centroid, the straightness of an edge is defined as

$$Cent_Edge = \begin{cases} 1, & \text{if } (C_x \in X) \cap (C_y \in Y) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

Using this definition, an edge is considered as a straight edge if ($Cent_Edge = 1$) or a curvise edge if ($Cent_Edge = 0$). Furthermore, each edge is divided into two sub-edges at the centroid, namely $Cent_SP1$ and $Cent_SP2$, to study the straightness property of the sub-edges. Using definition (3), $Cent_SP1$ and $Cent_SP2$ are classified as straight sub-edges or curvise sub-edges. Two more edge properties, namely $Edge_SP1$ and $Edge_SP2$ are obtained by dividing the edge into two equal-sized sub-edges and classifying these two sub-edges according to equation (3). In addition, the method finds the space between the edge components in the detected text blocks by counting the number of full blank columns in the detected text blocks. Finally, the method counts the number of edges satisfying the above conditions in the detected text block. The combinations of these features help in eliminating false positives. For example, the false positives in Figure 8(a) are eliminated using the condition (i). ((H

$< W) \& (N_V = 0) \& (N_{STCent} = 0) \& (N_S = 0))$ and (ii) $((H < W) \& (N_{EdgeSP2} = 0) \& (N_S = 0))$ where H is the height of the text block, W is the width of the text block, N_V represents number of vertical edges, N_{STCent} is the number of edges that satisfy $(Cent_Edge = 1)$, N_S is the number of space counts between the edge components and $N_{EdgeSP2}$ is the number of edges that satisfy $(Edge_SP2 = 1)$. The extracted true text blocks after elimination of false positives can be seen in Figure 8(b).



(a)Text blocks detection (b) Text blocks extraction

Figure 8. Sample false positive elimination

3. EXPERIMENTAL RESULTS

For experimental purpose, we created our own dataset as there is no standard dataset available in the literature. In this dataset, we included a variety of video images, including key frames containing texts taken from movies, news clips, sports videos and music videos. In total, 72 video images are considered for experimentation. There are both graphic text and scene text in the video images. The method implemented using MATLAB software is run on a PC with Pentium IV 2.33 GHz processor. The approximate processing time for each video frame is about 20 seconds which includes 2 seconds for initial text block selection, 11 seconds for text portions segmentation and 7 seconds for text detection.

3.1. Experiment on initial text block detection and text block separation

We use accuracy as the metric to evaluate the performance of the method of initial text block selection. The accuracy is defined as the number of images for which initial text block has been correctly chosen divided by the total number of images. The method successfully identifies initial text blocks for 68 images out of 72 images. Therefore, the accuracy is 94.4%. Figures 9(a) and 9(b) are two examples where the method fails to choose any initial text blocks due to very low contrast text in the images.

The proposed boundary growing method for text portion separation also fails for images such as Figure 9 (c) . Figure 9(d) shows the incomplete segmented text portion.



(a) (b) (c) (d)

Figure 9. Poor performance by initial text block selection and text portion separation methods

3.2. Experiment on text detection

The detected text blocks in an image are represented by their bounding boxes. To judge the correctness of the text blocks detected, we manually count the true text lines present in the image. We manually label each of the detected blocks as one of the following categories.

Truly detected text block is a detected block that contains partially or fully text. **Falsely detected text block** is a detected

block that does not contain any text. **Text block with missing data** is a detected text block that misses some characters.

Based on the number of blocks in each of the categories mentioned above, the following metrics are calculated to evaluate the performance of the method. **Detection rate** = Number of truly detected text blocks / Actual number of text blocks. **False positive rate** = Number of falsely detected text blocks / Number of detected text blocks (truly and falsely). **Mis-detection rate** = Number of text blocks with missing data / Number of truly detected text blocks. In addition, we also use processing time as a metric to evaluate the efficiency of the proposed method in comparing to existing methods.

Sample results of the proposed method and other methods are given in Figures 10 and 11. Method [10] is our recent method that introduces a method for elimination of non significant edges from the canny edge map of the segmented text portion, but this causes more misdetection and less detection rate as it eliminates sometimes text characters. Figures 10(a) (b) and 11(a) (b) compare the proposed method and our recent method. It can be seen that a few characters are missing in the last line in Figures 10(b) and 11(b). This results in more misdetection rate and less detection rate as reported in Table 3. On the other hand, the proposed method detects text blocks correctly without missing any characters as shown in Figures 10(a) and 11(a).

We have chosen three existing methods [7-9] for comparison. Method [7] makes use of uniform color for text location. Method [8] is based on the gradient for text detection. Method [9] is based on Sobel edge features. Figure 10 shows that the uniform color method (c) and edge based methods (e) fail to detect text blocks for the low contrast image whereas the gradient based method detects with inaccurate boundaries (d). Figure 11 shows that the uniform color(c), gradient (d) and edge based methods (e) fail to detect text blocks in the image successfully. Generally, edge based and gradient based methods suffer from the need to set numerous threshold values for text detection and hence the performance of those methods degrades compared to the proposed method. Thus our proposed method outperforms the existing methods in terms of the detection rate, false positive rate, and mis-detection rate as well.



(a) Proposed method

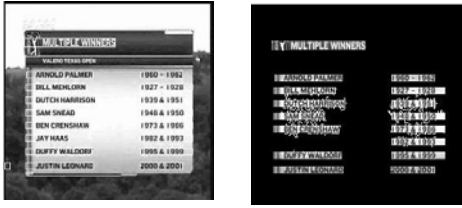


(b) Our recent method



(c) Uniform color (d) Gradient based (e) Edge based

Figure 10. Results on low contrast image



(a) Proposed method



(b) Our recent method



(c) Uniform color (d) Gradient based (e) Edge based

Figure 11. Results on sports video image

Table 1. Results of the proposed and existing methods

Methods	Text blocks	Detected	False positives	Misdetection
Uniform Color Method [7]	282	144	54	53
Gradient based Method [8]	282	200	27	20
Edge based Method [9]	282	225	60	45
Our recent method [10]	282	232	27	46
Proposed Method	282	268	18	25

Table 2 Performance of the proposed and existing methods

	Detection rate	False positive rate	Misdetection rate	Time in second
Uniform Color Method [7]	51.0%	27.2%	36.8%	42
Gradient based Method [8]	70.9%	11.8%	10.0%	26
Edge based Method [9]	79.7%	21.0%	20.0%	50
Our recent method [10]	82.2%	10.4%	19.8%	23
Proposed Method	95.0%	6.2%	9.3%	20

The results of comparison using the same dataset are summarized in Tables 1 and 2. We can see that the number of blocks detected by the proposed method is greater than the other four methods, because the proposed method detects text with small font and poor contrast while the other methods fail to do so. This results in a higher detection rate for the proposed method. Furthermore, the proposed method detects text blocks without missing characters. Thus the misdetection rate of the proposed method is lower than those of the existing methods. Processing time for each image

taken by the proposed and existing methods is reported in Table 2. Table 2 shows that the proposed method takes around 20 second for text detection in the image including initial text block selection and text portion segmentation, which is lower than the time taken by the existing methods because the proposed method does not involve any expensive methods for text detection in images.

4. CONCLUSION

In this paper, we propose a new inexpensive method for detecting both graphic text and scene text in video images accurately. The proposed method is based on segmentation of text portion and canny edge profiles. We have shown that text portion segmentation with its canny edge profile and false positive elimination combination performs better in text detection in images. The experimental results show that the proposed method outperforms the existing methods in terms of the detection rate, false positive rate, misdetection rate and processing time. The work will be continued to use of temporal and spatial information to detect text in complex background images.

5. ACKNOWLEDGMENT

This research is supported in part by IDM R&D grant R252-000-325-279.

6. REFERENCES

- [1] J. Zang and R. Kasturi. "Extraction of Text Objects in Video Documents: Recent Progress". *The Eighth IAPR Workshop on Document Analysis Systems (DAS2008)*, Nara, Japan, September 2008, pp 5-17.
- [2] D. Chen, J. Luetttin, K. Shearer. "A survey of text detection and recognition in images and videos". *Institute Dalle Molle, Intelligence Perceptive (IDIAP) research report*, IDIAP-RR 0038, August 2000.
- [3] K. Jung, K. I. Kim and A. K. Jain. "Text information extraction in images and video: a survey". *Pattern Recognition*, 37(5): 977-997, 2004.
- [4] A. K. Jain and B. Yu. "Automatic Text Location in Images and Video Frames". *Pattern Recognition*, 31(12): 2055-2076, 1998.
- [5] Y. Zhong, H. Zhang and A. K. Jain. "Automatic Caption Localization in Compressed Video". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(4): 385-392, 2000.
- [6] Q. Ye, Q. Huang, W. Gao and D. Zhao. "Fast and robust text detection in images and video frames". *Image and Vision Computing*, 23: 565-576, 2005.
- [7] V. Y. Mariano and R. Kasturi. "Locating Uniform-Colored Text in Video Frames". *IEEE 15th ICPR*, 4:539-542, 2000.
- [8] E. K Wong and M. Chen. "A new robust algorithm for video text extraction". *Pattern Recognition*, 36: 1397-1406, 2003.
- [9] C. Liu., C. Wang and R. Dai. "Text Detection in Images Based on Unsupervised Classification of Edge-based Features". *IEEE ICDA*, 2005, pp. 610-614.
- [10] P. Shivakumara, W. Huang and C. L. Tan. "Efficient Video Text Detection using Edge Features". *ICPR 2008*, December 8-11.
- [11] J. Zhang, D. Goldgof and R. Kasturi, "A New Edge-Based Text Verification Approach for Video", *ICPR 2008*, December 8-11.
- [12] R. C. Gonzalez and R. E. Woods, "Digital Image Processing", *Person Education*. 2002.