A Gradient Difference based Technique for Video Text Detection

Palaiahnakote 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 in video images has received increasing attention, particularly in scene text detection in video images, as it plays a vital role in video indexing and information retrieval. This paper proposes a new and robust gradient difference technique for detecting both graphics and scene text in video images. The technique introduces the concept of zero crossing to determine the bounding boxes for the detected text lines in video images, rather than using the conventional projection profiles based method which fails to fix bounding boxes when there is no proper spacing between the detected text lines. We demonstrate the capability of the proposed technique by conducting experiments on video images containing both graphics text and scene text with different font shapes and sizes, languages, text directions, background and contrasts. Our experimental results show that the proposed technique outperforms existing methods in terms of detection rate for large video image database.

1. Introduction

Since 1990s, with rapid growth of available multimedia and increasing demand for information indexing and retrieval, much effort has been done on text detection in video images [1]. A large number of approaches have been proposed and already obtained impressive performance under some constraints [1]. But detecting texts in video without any constraints remains challenging and interesting due to many undesirable properties of video images, such as low resolution, low contrast, unknown text color, size, position, orientation, color bleeding and unconstrained background [2,3]. Two types of text in video are: (1) c*aption/graphics/artificial* text which is artificially superimposed on the video by human, and (2) *scene* text which naturally occurs during video capture. Obviously, scene text detection is a challenging task compared to graphics text due to varying lighting, complex movement and transformation [1].

From the literature review it is realized that connected-component based methods are simple but

not robust because they are based on geometrical properties of components [4]. On the other hand, texture based methods may be unsuitable for small fonts and poor contrast text [5, 6]. In contrast to the preceding two approaches, edge and gradient based methods are fast and efficient but give more false positives when the complex background present [7-9]. However the major problem of these methods is in choosing threshold values to classify between text and non text pixels. A method based on uniform colors in L* a* b* space is also proposed in [10] to locate uniform colored text in video frames. Obviously, this method fails when text in video contains multiple colors in a text line or in a word. The above observation shows there is demand for developing a robust technique to give a better detection rate with fewer false alarms without any constraints for text detection in video images

Hence in this paper, we propose a new robust gradient difference technique for detecting text in video images. We observe that the high positive and negative gradient values exist nearer to text pixel or on text pixels compared with gradient of non text pixel. This observation motivated us to propose a gradient difference technique for text detection in video images. Further, instead of the conventional projection profile based method, we introduce a zero crossing technique for fixing boundaries of text lines in video images.
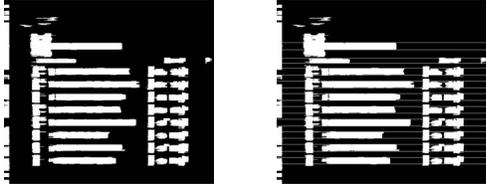
2. Text Detection Algorithm

2.1 Gradient Difference for Text Detection

It is noted in [9] that gradient information in text areas differs from non text regions because of high contrast of text. This is the basis of our gradient difference technique. For a given gray color image as shown in Figure. 1(a), the technique computes gradient *dx* image (G) by using a horizontal mask [-1 1] which gives rise to Figure 1(b). Then Gradient Difference (GD) is obtained for each pixel in G as the difference between the maximum and minimum gradient values within a local window of size 1×n centered at the pixel where n is a value that depends on the character's stroke width. In this study, we choose n = 11 by keeping small fonts in mind. High positive and negative gradient values in text regions result from

high intensity contrast between the text and background regions. Therefore, text regions will have both large positive and negative gradients in a local region due to even distribution of character strokes. This results in locally large GD values. To detect such large values the technique determines Threshold (T) automatically. It is shown in Figure. 1(c) where we can see text clearly as white patches and background as dark color. Small isolated white patches in Figure 1(c) are removed and the output is shown in Figure 1(d). Boundaries for white patches representing text lines are computed using a zero crossing technique which will be discussed in section 2.2, as shown in Figure 1(e). Figure 1(f) shows the text blocks detected and Figure 1(g) shows the extracted text blocks.
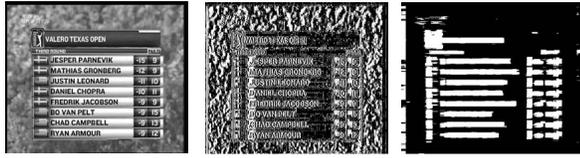
(a) Input image (b)Gradient image (c)Text segments

(d)Removed small white patches (e) Text boundary
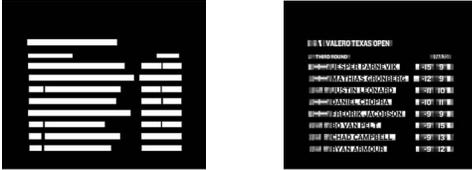
(f) Text blocks detected (g) Text blocks extracted

Figure 1. Text detection

More specifically, the algorithm for detecting texts in video images is as follows. Let F(x, y) be the given gray color image, G(x, y) be the gradient image obtained by convolving horizontal mask [-1 1] with F(x,y), and W(x, y) be the local window centered at (x,y) of size 1×11. Obtain the minimum and the maximum gradient values in W over G(x,y) as follows

$$Min\ (x, y) = \min_{x_i, y_i \in W(x,y)} (G(x_i, y_i)) \quad (1)$$

$$Max\ (x, y) = \max_{x_i, y_i \in W(x,y)} (G(x_i, y_i)) \quad (2)$$

Using equation (1) and (2), compute GD(x,y) as follows

$$GD\ (x, y) = Max\ (x, y) - Min\ (x, y) \quad (3)$$

Then a pixel is classified as follows

$$(x, y) = \begin{cases} Text\ \ pixel\ , & if\ (GD\ (x, y) > T) \\ Non\ \ Text\ \ pixel\ , & Otherwise \end{cases} \quad (4)$$

A global threshold (T) is determined based on the average value of gradient difference computed as follows. First we compute the average gradient values as:

$$AVG\ = \frac{1}{n \times m} \sum_{x=1}^{n} \sum_{y=1}^{m} G(x, y) \quad (5),$$

where n, m are the dimension of the gradient image. Next we count the number of High Gradient Values as
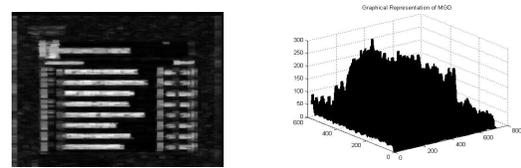
$$NHG\ = count\ (G(x, y) > AVG) \quad (6).$$

The sum of GD is computed as

$$SGD\ = \sum_{x=1}^{n} \sum_{y=1}^{m} GD\ (x, y) \quad (7),$$
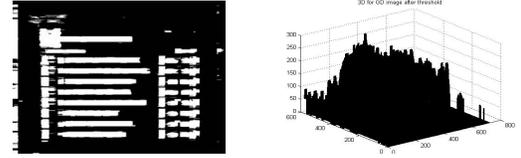
Finally the value of T is computed as

$$T\ = SGD \Big/ ((n \times m) - NHG) \quad (8).$$

Graphical representation for GD obtained by text detection algorithm before and after thresholding is given in Figure 2. It can be seen in Figure 2(d) that non text areas are suppressed by the threshold T.

(a) GD before thresholding   (b) 3D graph for (a)

(c) GD after thresholding   (d) 3D graph for (c)

Figure 2. Text and background separation

2.2. Zero Crossing for Fixing Bounding Boxes

Conventional projection profile based method fails to fix bounding boxes for text lines when there is no proper spacing between them. Figure 3(b) shows one such example where the second and third lines are connected to each other. These situations are common in video text detection. Therefore, we propose a zero crossing technique which does not require complete spacing between the text lines, to fix the boundary for such text lines. Zero crossing means transition from 0 to 1 and 1 to 0. The method counts the number of transitions from 0 to 1 and 1 to 0 in each column from top to bottom of GD(x,y). As shown in Figure 3(b) GD(x,y) is obtained for Figure 3(a) using the text detection algorithm given in section 2.1. Next it chooses the column which gives the maximum number

157

of transitions to be the boundary for the text lines. Here we ignore transition if the distance between two transitions is too small. With the help of the number of transitions, the technique draws horizontal boundaries for the text lines as shown in Figure 3(d). Further, the technique looks for spacing between the text components within two horizontal boundaries to draw the vertical boundary for the words and text lines. Then the detected text blocks are extracted as shown in Figure 3(e).

Lastly, in order to eliminate false positives, we compute height, width, aspect ratio, the number of Canny edges, the number of Sobel edges and the number of transitions from 0 to 1 and 1 to 0 in the detected text blocks. We eliminate the text blocks as false positives if the number of Canny edges is too little, or the number of transitions is too small or the absolute difference between the number of Canny edges and the number of Sobel edges is less than 2.
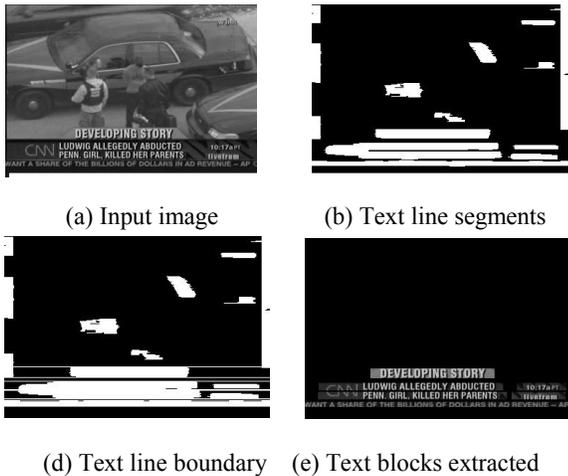

(a) Input image   (b) Text line segments

(d) Text line boundary   (e) Text blocks extracted

Figure 3. Advantage of zero crossing technique

3. Experimental Results

3.1 Dataset and Methods for Comparison

Since there is no benchmark database, we have created our own dataset for the purpose of experimentation. In this dataset, we have included a variety of video images such as images from movies, news clips (business, sports), news containing some scene texts, sports videos (golf, athletics), music video and web images. it also includes images of multiple languages such as English, Korean and Chinese. In this experiment, we have selected 488 video images from the above said sources which give 3231 actual number of text blocks. The method implemented using MATLAB software is run on a PC with Pentium 1V 2.33 GHz processor. The approximate processing time for each video image of size 352x288 is about 4 seconds for text detection.

We have chosen three existing methods [7, 9, 10] for comparison. Method [7] is based on Sobel edge information for text detection. Method [9] is based on gradient for text detection. However, as explained in section 1, these methods suffer from the choice of several thresholds. Method [10] makes use of uniform color for text location.

3.2 Sample Test Results

Figures 4-7 show the text detection results of the proposed method in (a) and the above three existing methods in (b)-(d) for a variety of sample video images. In (a), we show the original image, the text detection and the final text extraction results using the proposed method. In (b)-(d), we show only the text detection results of the three existing methods.

Figure 4 shows that the three existing methods fail for text detection in low contrast image whereas the proposed method detects most of the text in the image correctly.
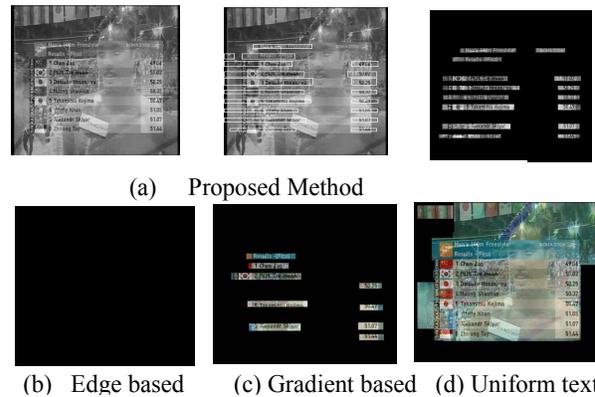

(a)   Proposed Method

(b)   Edge based   (c) Gradient based   (d) Uniform text

Figure 4. Text detection for low contrast image

Figure 5 shows that the proposed method detects both graphics and scene text in the athletic image with a false positive while the three existing methods fail to fix the text line bounding boxes correctly. The gradient based method fails to detect scene text but the uniform text color text method detects text successfully with a false positive.

Figure 6 shows that the proposed method detects text in the news image including small font present at the bottom of the image. While the edge and gradient based methods miss some text, the uniform text color method tends to include additional non text information in the bounding boxes. The gradient based method appears to detect small font and low contrast text better than the other two existing methods.

(a)   Proposed Method

(b) Edge based   (c) Gradient based (d) Uniform text

Figure 5. Text detection for athletic image

(a)   Proposed Method

(b) Edge based   (c) Gradient based (d) Uniform text

Figure 6. Text detection for news image

(a)   Proposed Method

(b) Edge based   (c) Gradient based (d) Uniform text

Figure 7. Text detection for complex background image

Figure 7 shows that both the proposed method and the edge based method detect text in complex background correctly. On the other hand, the gradient based method and uniform text color methods fail to detect text.

3.3 Comparison Metrics

We evaluate the performance of the proposed method by considering detection rate, false positive rate, misdetection rate and average processing time as decision parameters. The detected text blocks are represented by their bounding boxes. The Average Processing Time (APT) is measured for all images under study. To judge the correctness of the text blocks detected, we manually count Actual Text Blocks (ATB) in the images in the dataset. Also we manually label each of the detected blocks as one of the following categories:

Truly detected text block (TDB): a detected block that contains text fully or partially. **Falsely detected text block (FDB)**: a detected block that does not contain text. **Text block with missing data (MDB)**: a truly 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 techniques:

Detection rate (DR) = Number TDB / number of ATB. **False positive rate (FPR)** = Number of FDB / (number of TDB + number of FDB). **Misdetection rate (MDR)** = Number of MDB/ Number of TDB

The performance of the proposed technique in comparison with the existing methods is summarized in Table 1 and Table 2. Table 2 shows that the detection rate of the proposed method is higher than the three existing methods. Compared with the existing gradient method, the present method degrades somewhat in the false positive rate and misdetection rate. This is insignificant considering the much higher detection rate of the present method. The average processing time of the present method is also comparable to the existing gradient method.

Table 1: Results based on experimental study for the proposed and existing methods

Method	ATB	TDB	FDB	MDB
Edge based [7]	3231	1288	112	217
Gradient based [9]	3231	1368	116	0
Uniform text color [10]	3231	1996	379	1035
Proposed	3231	3085	212	63

Table 2: Performance (%) of the proposed and Existing methods based on values reported in Table 1

Method	DR	FPR	MDR	APT(sec)
Edge based [7]	39.8	8.0	16.8	25
Gradient based [9]	42.3	**7.0**	**0**	**3**
Uniform text color [10]	61.7	15.9	51.8	42
Proposed	**95.4**	9.3	2.0	4

3.4 Experiment on window size

We have conducted experiments for the image shown in Figure 6(a) to choose proper n which we used in section 2.1 for detecting text candidates using gradient difference values as shown in Figure 8.

(a) n = 4　　　(b) n = 9　　　(c) n = 11
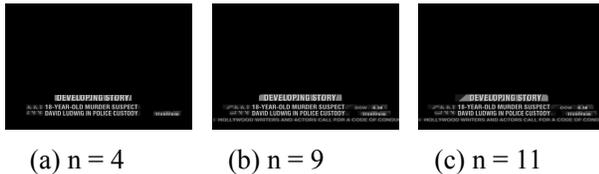
Figure 8. Choosing n values

It is noticed from Figure 8 that for n = 4 we lost low contrast text bottom line, for n = 9, it restore bottom line but it misses right side low contrast text and for n = 11, the method detects all text lines. Hence we choose n = 11 in this work. Further, it is also noticed in Figure 8(b) that first line looks like cropped whereas in (c) text line restored completely.

3.5 Limitation of the proposed Method

Despite its better performance than the existig methods, the proposed method has a limitation in that it fails to fix bounding boxes for staggered text lines or skewed scene text as shown in Figure 9. Solution to this problem will be handled in future.

(a)Staggered text lines　　　(b) Skewed scene text

Figure 9. Failure in fixing bounding boxes by the proposed method

4. Conclusion and Future Work

In this paper, we propose a gradient difference based text detection technique for extracting both graphic text and scene text with different fonts, size, scripts, contrast, orientation and backgrounds. A zero crossing technique for fixing bounding boxes for touching text lines is proposed rather than the projection profile based method. Experimental results showed that the proposed method gives good detection rate comparing with the results of three existing methods.

For our future work, we plan to use temporal information to reduce the false positive rate and misdetection rate because temporal information will help in locating exact text position in the video images. Furthermore, the method can be extended to fix the bounding boxes for text lines with arbitrary direction by considering the detected text block as seed point to trace the direction of the remaining text portion.

4. 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]　K. Jung, K.I. Kim and A.K. Jain. "Text information extraction in images and video: a survey". *Pattern Recognition,* 37, 2004, pp. 977-997.

[3]　Q. Ye, Q. Huang, W. Gao and D. Zhao. "Fast and robust text detection in images and video frames". *Image and Vision Computing* 23, 2005, pp. 565-576.

[4]　A.K. Jain and B. Yu. "Automatic Text Location in Images and Video Frames". *Pattern Recognition*, Vol. 31(12), 1998, pp. 2055-2076.

[5]　Y. Zhong, H. Zhang and A.K. Jain. "Automatic Caption Localization in Compressed Video". *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 22, No. 4, 2000, pp. 385-392.

[6]　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 Transactions on Pattern Analysis and Machine Intelligence,* Vol. 25, No. 12, December 2003, pp 1631-1639.

[7]　C. Liu, C. Wang and R. Dai. "Text Detection in Images Based on Unsupervised Classification of Edge-based Features". *ICDAR 2005*, pp. 610-614.

[8]　P. Shivakumara, W. Huang and C. L. Tan." An Efficient Edge based Technique for Text Detection in Video Frames". *The Eighth IAPR Workshop on Document Analysis Systems (DAS2008)*, Nara, Japan, September 2008, pp 307-314.

[9]　E. K. Wong and M. Chen. "A new robust algorithm for video text extraction". *Pattern Recognition* 36, 2003, pp. 1397-1406.

[10] V. Y. Marinano and R. Kasturi. "Locating Uniform-Colored Text in Video Frames". *15th ICPR,* Volume 4, 2000, pp 539-542.