2012 10th IAPR International Workshop on Document Analysis Systems

A Fast Stroke-Based Method for Text Detection in Video

Bo Bai, Fei Yin, Cheng-Lin Liu
National Laboratory of Pattern Recognition
Institute of Automation of Chinese Academy of Sciences
95 Zhongguancun East Road, Beijing 100190, P. R. China
{bbai, fyin, liucl}@nlpr.ia.ac.cn

***Abstract*—Texts in video provide a rich clue for video indexing and retrieval, yet the detection and recognition of video text remains a challenge. This paper proposes an effective and real-time stroke-based method for text detection in video, which is robust to the change of stroke intensity and width. Particularly, we propose to characterize the text confidence using an edge orientation variance (EOV) and an opposite edge pair (OEP) feature. Based on the text confidence map, candidate text components are extracted and grouped into text lines by thresholding and connected component analysis. Our experimental results demonstrate that the proposed method can detect multilingual texts in video with fairly high accuracy.**

Keywords – video text detection; edge-based; stroke-based; stroke edge pair feature

I. INTRODUCTION

With the rapid development of TV, Internet, and wireless network, the demand of video indexing and retrieval is increasing. The texts in video provide an informative clue for video indexing and retrieval because texts carry semantic information more relevant to the video contents than images. However, the detection and recognition of texts in video is a challenging problem because of the variable font type and color of texts and the cluttered background. Despite the numerous efforts reported so far [1-16], this problem is not solved yet.

Text detection and location is the first step of image and video text information extraction, and has received high attention in research. The proposed methods so far can be roughly categorized into three groups [3][4]: connected-component (CC)-based, texture-based and edge-based ones.

CC-based methods [5] generally assume uniform color in the characters and extract character candidate components by image segmentation using color or gray intensity uniformity. The candidate components are then verified according to character shapes and spatial context. However, the extraction of character components is not trivial in cluttered images due to the variability of character color and illumination.

Texture-based methods [6] assume that text regions in images and videos have distinct textural properties from the background. Usually, candidate regions of variable scales are scored using a binary classifier on extracting textural features (such as Gabor filters, wavelet transform, gradient orientations, local binary pattern (LBP) features). The classification of a large number of candidate regions makes texture-based methods computationally expensive.

Edge-based methods [1][2][7], taking advantage of the rich edge information of text regions, have been adopted for fast text detection. On edge detection of the whole image using, e.g., the Sobel or Canny operator, some strategies are used to enhance the text edges and inhibit the background edges. Then, text edges are grouped into text regions, often using morphological operators. This method may detect text of variable size without need of multi-scale scanning of candidate regions, and therefore, is computationally efficient. However, its performance relies on the extraction of text edges, which are often contaminated by background edges.

To better distinguish text regions from the background, some methods have utilized the stroke characteristics of texts [8-12]. A distinct characteristic of strokes is that they have approximately uniform width and double edges of opposite gradient directions. Ye et al. [8] calculated the double-edge strength of gray scale image based on the oriented two-sided intensity contrast. Jung et al. [9][10] proposed a stroke filter to generate a stroke map based on the oriented two-sided contrast as well as the homogeneity of central region and lateral regions of each pixel. The assumption of stroke intensity homogeneity, however, is not observed in many images due to the illumination change. This method also suffers from high computation because of the hypothesized stroke width in a wide range. Epshtein et al. [11] proposed the so-called Stroke Width Transform (SWT), which searches for each pixel the stroke width along the direction of gradient. Despite its promise of text candidate region filtering, the search of stroke width for all pixels is computationally demanding and is likely to be sensitive to image noises.

In this paper, we propose a fast stroke-based method for text detection in video images. Particularly, we propose to characterize the text confidence of image using an edge orientation variance (EOV) and an opposite edge pair (OEP) feature. These features are calculated in local regions efficiently and are robust to the changes of stroke intensity and stroke width. Also, the features are independent of languages, and so, they enable our method to detect multilingual texts. Our experimental results on a large video dataset demonstrate the effectiveness and efficiency of the proposed method.

The rest of this paper is organized as follows. Section 2 describes the proposed text detection method; Section 3 presents the experimental results, and Section 4 provides concluding remarks.

978-0-7695-4661-2/12 $26.00 © 2012 IEEE
DOI 10.1109/DAS.2012.3

II. PROPOSED METHOD

On observing a great deal of edge maps in scene text and video text images, we summarize three prominent characteristics of text regions. First, the test regions have dense and strong edge pixels. Second, the edge orientations of text region are highly variable. Third, because characters are made up of strokes, the stroke edges always appear in pairs with opposite gradient direction. Inspired by the above observations, we devise our text detection method by extracting features highlighting these text and stroke characteristics.

The block diagram of our video text detection system is shown in Fig. 1. The input to the system is a video image sequence, and the output is a sequence of text strings. We sample one frame per three, and each sampled frame is processed by edge detection, text confidence map (TCM) generation, and candidate text region detection. Moreover, the final text region image is refined by multiple frame integration (MFI).

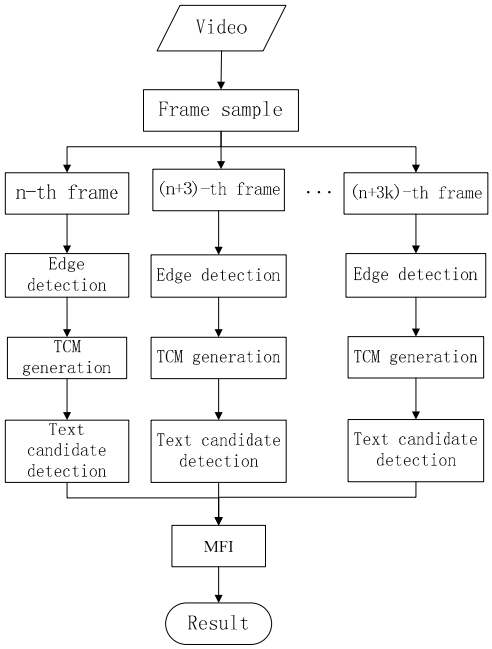

Figure 1. Block diagram of the proposed method.

A. Edge detection

The richness of edges is an effective and efficient characteristic of text regions and have widely adopted for text detection. The edge strength is a widely used feature and have shown promising performance [1][2]. Moreover, due to the variable shapes of characters and strokes, the edges in text regions often show high variance of orientations (usually quantized into four orientations: horizontal, vertical, up-right and up-left) [1].

In addition to edge strength and variance of orientations, our method makes use of more characteristics of text edges. It has been observed that strokes not only have multiple orientations, but also have nearly constant width (though it is unknown a priori) [9][11][12]. The double edge with opposite gradient directions has also been observed and utilized in text detection [10-12]. We call this as opposite edge pair (OEP) (Fig. 2) and measure it in a computationally efficient manner.

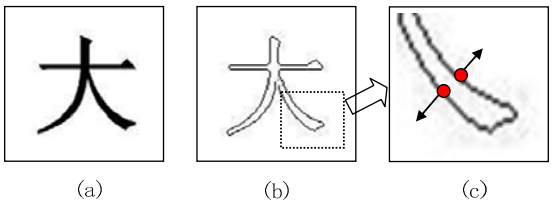

Figure 2. Illusrtation of OEP: (a) original image, (b) edge image, (c) OEP (red edge points).

As a pre-processing step, we first extract the edges of each frame image using the Sobel operator; calculate the strength and direction of gradient for each edge pixel. The gradient direction is quantized into eight directions, represented by two parameters θ, λ as shown in Fig. 3. θ represents the quantized orientation of edge pixels ($\theta \in \{0,45,90,135\}$); λ represents polarity of orientation ($\lambda \in \{+1,-1\}$, "+1" denotes up-right, "-1" denotes down-left).

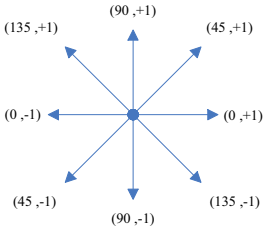

Figure 3. Edge gradient directions represented by (θ, λ).

B. Text Confidence Map Generation

The text confidence map (TCM) is a gray-scale image with the same size as the input image, measuring the probability that each pixels belongs to text region. Based on the TCM, text candidate regions detection and grouping become easier than that on the original image. We generate the TCM taking into account three features based on edges: edge density, variance of edge orientation and the number of OEP. The last feature, OEP number, characterizes the nature of strokes.

The density of edges can discriminate well text regions and simple backgrounds. We use the edge density calculated based on the average edge strength within a window [1]:

$$D(x_0, y_0) = \sum_{x=-a}^{a} \sum_{y=-b}^{b} edge(x_0 + x, y_0 + y), \qquad (1)$$

where $edge(x, y)$ is the edge strength of a pixel (x, y), a and b are the width and height of the window, respectively.

From our observation, the variance of edge orientations in a local region is an indicator for differentiating text edges

and non-text edges: text edges are more variable in orientation because of variable stroke shapes. We calculate the edge orientation variance (EOV) feature as

$$f_{EOV}(\delta) = -\sum_{\theta}\left(\frac{4\sum_{\lambda} n(\theta,\lambda,\delta) - N(\delta)}{3N(\delta)}\right)^2, \qquad (2)$$

where δ is a window centered at (x_0, y_0) of size $a \times b$, $n(\theta,\lambda,\delta)$ is the number of edge pixels of direction (θ,λ) within the window δ, and $N(\delta)$ is the total number of edge pixels in the window. In equation (2), when $\sum_{\lambda} n(\theta,\lambda,\delta)$ equals $\frac{1}{4}N(\delta)$ (namely, the region contains the same number of edge pixels in four orientations), $f_{EOV}(\delta)$ reaches its maximum value which means that the region is more likely to be a text region. Fig. 4 shows the effect of $f_{EOV}(\delta)$. In this example, if we only consider the edge density, a high-density no-text region would be falsely detected as candidate text region.

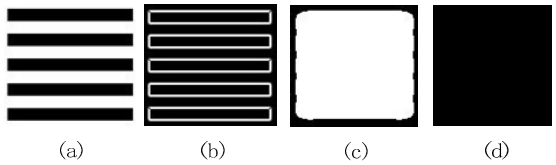

Figure 4. Example to show the effect of f_{EOV} :(a) the original image, (b) the edge of original image, (c) the text region only the edge density being considered (d) the text region both edge density and f_{EOV} being considerd

Since the strokes have constant width, edge points of a stroke always appear in pair and the two edge points in one pair have opposite directions. In order to manifest this feature in text region, we introduce the feature of OEP number (f_{OEP}) to enhance text confidence in text region. It is described as (3).

$$f_{OEP}(\delta) = \sum_{\theta} f_{oep}(\theta,\delta), \qquad (3)$$

$$f_{oep}(\theta,\delta) = \begin{cases} -\dfrac{\phi(\theta,\delta)}{\varphi(\theta,\delta)} & \varphi(\theta,\delta) \neq 0 \\ t_3 & \varphi(\theta,\delta) = 0 \end{cases}, \qquad (4)$$

$$\phi(\theta,\delta) = \left| n(\theta,+1,\delta) - n(\theta,-1,\delta) \right|, \qquad (5)$$

$$\varphi(\theta,\delta) = n(\theta,+1,\delta) + n(\theta,-1,\delta), \qquad (6)$$

where $\phi(\theta,\delta)$ and $\varphi(\theta,\delta)$ are defined as (4) and (5), respectively. t_3 is a predefined threshold (smaller than the minimum of $-\frac{\varphi(\theta,\lambda)}{\phi(\theta,\lambda)}$) to cover the condition that the number of edge pixels in one orientation is zero which indicates that the region is less like a text region.. In formulation (3), the $f_{OEP}(\delta)$ is a monotonic function, the larger the function value is, the more a region likes a text region. Fig. 5 shows the effect of this factor. In this example, if we only consider the edge density and direction, a high-density and multi-edge-direction but no-text region would be falsely detected as candidate text region.

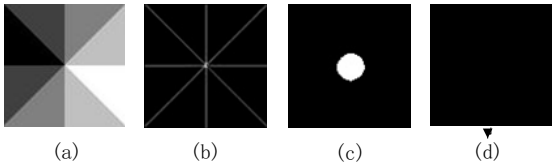

Figure 5. Example to show the importance of f_{OEP} :(a) is the original image, (b) is the edge of the image, (c) is the text region only the edge density and f_{EOV} being considered (d) is the text region edge density f_{EOV} and f_{OEP} all being considerd

According to above three features of stroke, we can define our text conference map as:

$$TC(x,y) = D(x,y)\exp\left[f_{EOV}(\delta) + f_{OEP}(\delta)\right], \qquad (7)$$

Fig. 6 shows an example how to generate text conference map. From fig. 6(c)-(h), we can see that f_{EOV} and f_{OEP} can effectively filter out no-text region from text conference map.

C. Candidate Region Generation

Since the intensity of the text confidence map represents the possibility of text, OTSU algorithm and connected component analysis (CCA) are used to get highlight candidate text region. Then, each candidate region is enclosed in a boundary box (text box). Though our text confidence map is fairly accuracy, there still exist a few falsely alarm blocks. Therefore, two constraints are used to filter out those text blocks which are too small to contain text. These two constraints are defined as follows:

$$\min(text\_box\_width, text\_box\_height) < t_1, \qquad (8)$$

$$\max(text\_box\_width, text\_box\_height) < t_2, \qquad (9)$$

where t_1 and t_2 are predefined threshold, and the values of them are determined by the size of text in the video frame.

D. Text Region Refinement

Normally, the text in the video must last for a few seconds. Hence, we can continue to use this property to filter out non-text regions and enhance text regions [7] [13] [14] [15].

In order to calculate the duration of a candidate text, we need to get its beginning frame and ending frame. In our system, when a candidate text region is firstly detected in a frame, we defined this frame as its beginning frame. After getting the beginning frame, whether the following frames is the ending frame is judged by following two steps.

The first step is to make sure whether there is a candidate text block with similar size and position in the current frame as in the former frame. If the answer is negative, the current frame is the ending frame and we could get its duration without going to the second step. On the contrary, we

continue to the second step. In this step, we need to ensure whether the caption is the same. The method we used is the edge points matching approach as described in [15]. If the answer is negative, it means this text has finished and can obtain its duration. Oppositely, it represents the text doesn't end in the current frame and goes to the first step.

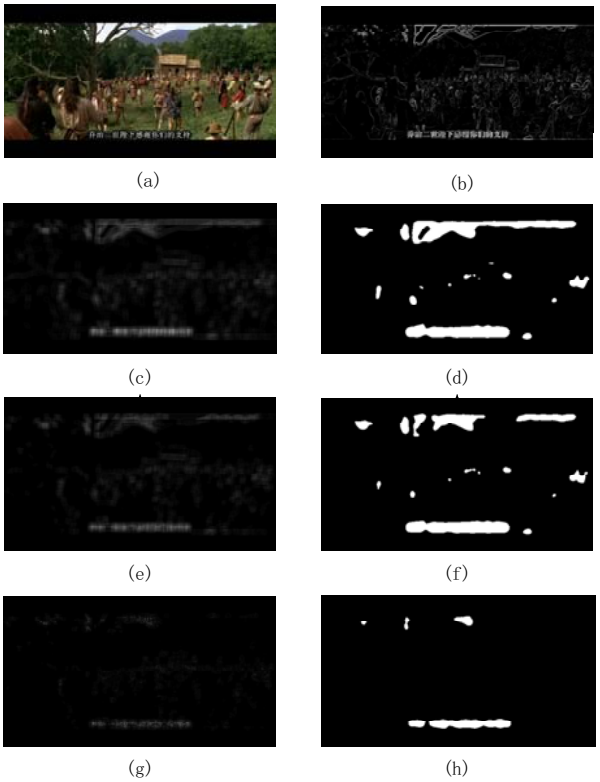

Figure 6. Text confidence text: (a) is the original image, (b) the edge of the image, (c) (d) the text confidence map obtained by D and the corresponding binaryzation image,, (e) (f) the text confidence map obtained by D and f_{EOV} and the corresponding binaryzation image, (g) (h) the text confidence map obtained by D, f_{EOV}, and f_{OEP} and the corresponding binaryzation image,.

After we get the duration of a candidate text, if the duration is less than a predefined threshold t_4, we think this candidate text is a noise and delete it [16]. Therefore, the survival texts of this step are the real texts (Each real text is a series of text blocks with the same caption and similar size). The flow chart of this procedure is described as Fig. 7.

Because the text is stable and background (no-text region) is variance in a series of frames, we enhance the text blocks by average integration [5] (described as (10)) to get more clear text region.

$$final\_text\_region(x,y)=\frac{1}{N}\sum_{i=1}^{N}text\_region_i(x,y), \quad (10)$$

where N is the number of candidate text blocks in the series frame. $text\_region_i$ is the i-th text block.

Since the text region which we obtain by MFI is clear, a sample edge horizontal and vertical projection method can be used to segment text lines and get the final text region. Fig. 8(a)-(c) show three text blocks in a series, and Fig. 8(d) shows an example of the result of text region refinement.

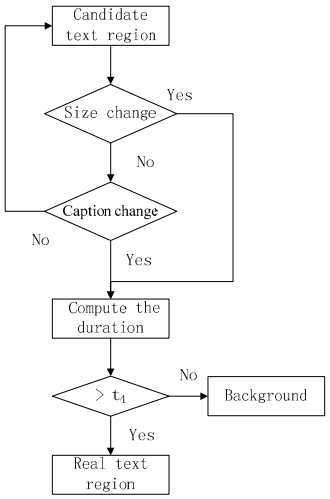

Figure 7. Flow chart of text region refinement.

Figure 8. Example of text region refinement: (a)-(c) three text blocks in a series, (d) the result of MFI.

III. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed approach, we took the experiments on CASIA-MOVIE database [17]. There have been some other datasets for video text detection in the literature, but they are either very small or not publicly available. CASIA-MOVIE database belongs to CASIA-VDB database and includes six video clips containing Chinese, English and a few digital texts, in which the size of video frame are 624*352, 720*416, 576*304, and 608*366, we normalize the height of all the video frames to height of 352 pixels and proportional to width for obtaining the computation time equitable. Here, we divide the experiments into two phases. The first phase focuses on the validity of f_{EOV} and f_{OEP}. The second phase focuses on the performance of our method.

Our experiments were implemented on a PC with Intel(R) Core(TM)2 Duo CPU-3.0GHz and programming language was C++.

We measure the text detection performance using four metrics: area-based recall (R_A), area-based precision (P_A), line-based recall (R_L) and line-based precision (P_L). They are defined as follows:

$$recall = \frac{P_d \bigcap P_g}{P_g}, \tag{11}$$

$$precision = \frac{P_d \bigcap P_g}{P_d}, \tag{12}$$

$$line\_recall = \frac{N_d \bigcap N_g}{N_d}, \tag{13}$$

$$line\_precision = \frac{N_d \bigcap N_g}{N_g}, \tag{14}$$

where P_d is the size of the detected region, P_g is the size of the ground-truth region. If the recall of a text line is more than 0.9, we define this text line has been detected. N_d is the number of the text line which has been detected, and N_g is the total number of the text line.

In the first experiment, we focused on the effective of EOV feature (f_{EOV}) and OEP feature (f_{OEP}). Hence, we choose 176 frames from all 14804 frames for test (for the frames of the same text, only one was selected). Furthermore, because all the results were gotten on single frame, MFI was not included. The results were shown in Table 1, where D is edge density. From the table, we can see that these two factors can improve performance significantly.

TABLE 1. EVALUATION OF TEXT DETECTION

	R_A	P_A	R_L	P_L
D	0.9185	0.7955	0.9182	0.5563
$D + f_{EOV}$	0.9217	0.7940	0.9182	0.6955
$D + f_{EOV} + f_{OEP}$	**0.9256**	**0.8142**	**0.9257**	**0.9356**

In the second experiment, we do experiments on the six video clips. In our system, we did detection operation once every three frames, and parameters are set as: $a = 7$, $b = 7$, $t_1 = 15$, $t_2 = 40$, $t_3 = -2$, $t_4 = 3$. The time for processing one frame is about 0.034 seconds, and the system achieved line- recall of 0.9442 and line-precision of 0.9621 on the test sets.

IV. CONCLUSIONS

This paper presents a novel stroke-based method for text detection in video, in which the edge strength, variance of orientations and OEP are all used to make the method more robust to multi-language, complex background, various direction, font and color of the text. Experimental results show the efficiency of our approach with the line-recall rate of 0.9257 and line-precision of 0.9356 on frame images. After multiple frame integration, we obtained line-recall rate of 0.9442 and line-precision of 0.9621 on video data. In the future, we will continue to introduce more robust stroke-based feature into our method to improve the performance of our system.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 60825301 and Grant 60933010, National Basic Research Program of China (973 Program) Grant 2012CB316302.

REFERENCES

[1] X.-Q. Liu and J. Samarababdu, "Multiscale edge-based text extraction from complex images," *Proc. Int'l. Conf. on Multimedia and Expo (ICME 06)*, 2006, pp. 1721.

[2] M. R. Lyu, J. Q. Song, and M. Cai, "A comprehensive mehtod for multilingual video text detection, localization, and extraction," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 15, No. 2, February 2005, pp. 243–255.

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

[4] D. Chen, J. Luettion, K. Shearer, "A survey of text detection and recognition in images and videos," *Institut Dalle Molled'Intelligence Artificielle Perceptive (IDIAP) Research Report,* IDIAP-RR 00-38, Aug. 2000

[5] K. C. Jung, J. H. Han, K. I. Kim, and S. H. Park, "Support vector machines for text ocation in news video images," *Proc. IEEE Region 10 Conf. System Technology Next Millennium*, vol. 2, 2000, pp.176-180.

[6] W. Mao, F. Chung, K. Lanm, W. Siu, "Hybrid chinese/english text detection in images and video frames," *Proc. Int'l. Conf. on Pattern Recognition*, Vol. 3, Quebec, Canada, 2002, pp. 1015-1018.

[7] V. C. Dinh, S. S. Chun, S. Cha, H. Ryu, and S. Sull, "An efficient method for text detection in video based on stroke width similarity," *Proc. Asian Conf. on Computer Vision (ACCV 07)*, 2007, pp. 200-209.

[8] X. Ye, M. Cheriet, C.Y. Suen, "Stroke-model-based character extraction from gray-level document images**,"** *IEEE Trans. Image Processing*, vol.10, no.8, pp.1152-1161, 2001.

[9] Q. Liu, C. Jung, and Y. Moon, "Text segmentation based on stroke filter," *Proc. Int'l. Conf on Multimedia (MM 06)*, 2006, pp. 129-132.

[10] C. Jung, Q. Liu, and J. Kim, "A stroke filter and its application to text localization," *Pattern Recognition Letters*, vol. 30, 2009, pp. 114–122.

[11] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," *Proc. Int'l. Conf. on Computer Vision and Pattern Recogintion (CVPR 10)*, 2010, pp. 2963-2970.

[12] J. Zhang and R. Kasturi, "Character Energy and Link Energy-Based Text Extractio in Scene Images," *Asian Conf. on Computer Vision (ACCV 10)*, 2010, pp.308-320.

[13] F. Chang, G. C. Chen, C. C. Lin, and W. H. Lin, "Caption analysis and recognition for building video indexing systems," *Multimedia Systems*, vol. 10, 2005, pp. 344-355.

[14] X. S. Hua, P. Yin, and H. J. Zhang, "Efficient video text recognition using multiple frame integration," *Proc. Int'l Conf. on Image Processing (ICIP 04)*, Sep. 2004, pp. 22-25.

[15] C.-J. Liu, C. Liu, H, Chen, "A simple method for Chinese video OCR and its application to question answering," *Computational Linguistics and Chinese Language Processing*, vol. 6, No. 2, 2001, pp. 11-30.

[16] W. Kim and C. Kim, "A new approach for overlay text detection and extraction from complex video scene," *IEEE Trans. on image processing*, vol. 18, No.2, Feb. 2009, pp. 401-411.

[17] X.-F. Wang, "Video OCR Research," Ph.D thesis, Institute of Automation Chinese Academy of Sciences, 2011.