Robust Scene Text Detection Based on Color Consistency

Yang Zheng[1], Heping Liu[1], Jie Liu[2] , Qing Li[1], Gen Li[2]
1 University of Science and Technology Beijing, China
2 Institute of Automation, Chinese Academy of Sciences, China

ABSTRACT

The whole process of text detection in scene images always contain three steps: character candidate detection, false character candidate removal, words extraction. However some errors appear in each step and influence the performance of text detection. According to the disadvantages of each step, we propose the compensation methods to solve these problems. Firstly, a filter based on color of stroke named Stroke Color Transform is used to ensure the integrality of characters and remove some false character candidates. Secondly, a classifier is trained based on gradient features is adopted to remove false character candidates. Thirdly, an extractor based on color of consecutive character named Character Color Transform is employed to extract undetected characters. The proposed technique is test on the two public datasets i.e. ICDAR2011 dataset, ICDAR2013 dataset, the experimental results show that our approach outperforms the state-of-the-art methods.

Keywords: text detection, Stroke Color Transform, gradient feature, Character Color Transform

1. INTRODUCTION

Texts in scene images and videos contain rich high-level semantic information, text detection has attracted attentions more and more in the computer vision community in recent years, it is important in many practical applications such as multilingual translation, image retrieval, object recognition. However, scene text detection is nontrivial due to background clutters, illumination changes, the variation of text position, font, color and line orientation. In order to obtain text information from scene images, two stages are usually included: text detection and text recognition. The text recognition is based on the text detection, so we focus on scene text detection stage in this paper.

There are two mainly approaches used in text detection: those based on a sliding window and another based on connected component extraction. Sliding window methods usually train discriminative models to detect text at multiple scales [1, 2]. Image patches are classified by other models, such as texture, shape or appearance models, which are then grouped into text regions. However, in the clutter backgrounds and multilingual environments, direct patch discrimination cannot detect image patch correctly, because a small image patch often does not contains sufficient discriminative information. Connected component extraction methods often use color [3], stroke [4, 5], edge/gradient [6], region [7, 8, 9] features or a combination of them [10, 11, 3, 12] to detect characters or character parts, which are then grouped for further classification. Maximally Stable Extremal Regions (MSER) [13] and Stroke Width Transform (SWT) [4] are the representation of this method. The algorithm based on the MSER has attracted more and more attentions in recent years. MSER algorithm can adaptively detects stable color regions as text components. Despite of the effectiveness of MSER method in text localization, it also require additional cues for text/non-text discrimination. Stroke analysis is a preferred component based method for text localization, stroke is the basis of some subsequent works, Epshtein [4] proposed a novel algorithm named Stroke Width Transform (SWT), this method has been shown effective and some recent approaches are based on SWT. Typical stroke based text detection approaches [14] uses regions of strokes as text candidates. Weilin Huang [15] proposed a Stroke Feature Transform (SFT) filter based on SWT and trained two classifier to identify characters and words, the SWT algorithm also serves as the foundation of the character candidate detection which proposed in this work.

The remainder of the paper is organized as follows: The text detection approach is described in Section 2. Experiments are presented in Section 3 and we conclude the paper in Section 4.

2. TEXT DETECTION APPROACH

In this section, we describe the proposed method in detail, the proposed text detection system is organized four main steps: (1) character candidate detection; (2) character candidate verification; (3) undetected character extraction; (4) word generation. Figure 1 shows the pipeline of the proposed system, and each stage will be described in detail as follows, as well as the sample results of each stage is presented in Figure 2:

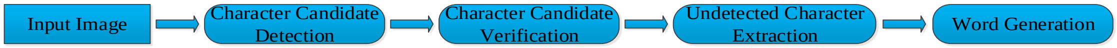
Input Image
Character Candidate Detection
Character Candidate Verification
Undetected Character Extraction
Word Generation

Figure 1 Flowchart of the proposed system

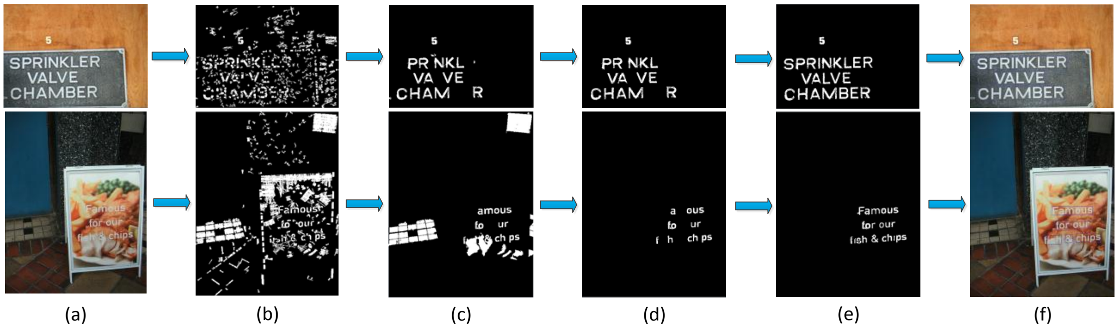

Figure 2 Results after each stage of the sample. The input images are shown on the image (a), the Stroke Width Transform is used to detect connected components are shown on the image (b), the results of SCT filter are shown on the image (c), the result of character candidate verification are shown on the image (d), the results of undetected character extraction are shown on the image (e), (f) is the final results.

2.1 Character Candidate Detection

The recently introduced Stroke Width Transform has been shown to be effective for text detection in the wild. But it has two disadvantages: a character is separated into some components due to the irregular orientations point to the outside of character; some pixels that are not edges of character will be merged into a connected component if the relation of gradient orientations between two pixels satisfy the threshold.

The latter parts of our system solely rely on the output of this step, so remedy these problems is very critical in the overall system. According to the principle of neighborhood coherency constraint, we use the color information to refine connected component. We refer to this new method as Stroke Color Transform (SCT). The process of SCT filter in detail as follows:

1) Connected components are detected by SWT.

2) Compute the means and variances of pixels in component. The relationship between color values, means and variances in R G B channels is described in the following.

$$Color_{mean} - 3Color_{\sigma^2} \le Color \le Color_{mean} + 3Color_{\sigma^2} \tag{1}$$

3) If at least two color values of pixel that surround the pixels of connected component satisfy the Equation 1, this pixel will be added to the connected component, repeat the process until no pixel satisfies the condition, the final component is named expanding component.

4) If the ratio of the width or height between original component and expanding component satisfy a certain proportion simultaneously, the expanding component will be considered as a character candidate to replace the connected component, otherwise, two components are removed.

It is clear that some characters are refined correctly and some false components are removed after filtering of SCT in Figure 2(c). The result shows that it increases the number of character candidates and reduces the number of false components.

2.2 Non-character Filtration

In order to remove non-character effectively, this stage involves two operations. We briefly describe it in this part, which mainly follows the previous work in [16]:

1) The realization of normalization. The input image is transformed into the normalized image;

2) Feature extraction technique. Firstly, the normalized image is blurred. Secondly, compute the x and y components of gradient and the gradient is decomposed into eight directions corresponding to eight feature planes. Thirdly, compute the features based on the eight feature planes.

A SVM classifier is trained by using features to remove non-characters, the results are described in Figure 2(d).

2.3 Undetected Character Extraction

Some characters named undetected characters are not detected in the previous steps, the color information is used to extract them based on the color consistency of consecutive characters. We refer to this new method as Character Color Transform (CCT), the related work of CCT extractor in detail as follows:

1) Some properties such as color values, character width, character height and spaces between characters and words are used to confirm characters whether they are in a line or not.

2) The undetected characters always exist in the three areas in a text line: the left area of text line, the area between consecutive characters and the right area of text line.

For the area between consecutive characters, if the distance between consecutive characters exceed 0.5 times the width of the wider one, we will detect the undetected characters from left to right, three steps for detecting undetected characters are following:

Firstly, confirm the initial pixel based on color information. The left character is considered as basic character and the means and variances of pixels in basic character are computed. Search a pixel from bottom to up to right at the three pixels distance, if they are satisfied the Equation 1 in R G B channels, the pixel will be considered as initial pixel and put it into a set named pixelsChar, otherwise, continue to move three pixels distance to right until find the initial pixel or greater than the right character.

Secondly, search other pixels based on the initial pixel to construct a connected component. If color values of two channels at least of pixel that surrounds the pixels of pixelsChar satisfy the Equation 1, the range of color value of other channel expand ten values, this pixel will be pushed into pixelsChar and continue to search other pixels repeat above process, so the undetected character candidate will be made up by the pixels of pixelsChar.

Thirdly, combine the geometry to identify the undetected character candidate. The rules based on geometry as follows: the number of pixels in the pixelsChar; the width ratio and height ratio between undetected character candidate and basic character; the vertical distance of two characters; the character candidate cannot cover half of the basic character. If undetected character candidate satisfies above rules, it will be the undetected character and be considered as the basic character to extract other undetected characters. Otherwise, move three pixels distance to right to search the new initial pixel.

The methods of extracting undetected characters in the left area of text line and right area of text line are very similar with mentioned above, some results of CCT extractor are illustrated in Figure 2(e).

2.4 Word Generation

In order to separate text lines into words, we use a heuristic that computes a histogram of horizontal distances between consecutive characters and estimate the distance threshold that separates intra-word characters from inter-word characters. Firstly, we compute the distances between consecutive characters. Secondly, the distance threshold is computed based on the median of distances. If the distance between consecutive characters is less than threshold, they will be clustered together into a word.

3. EXPERIMENTS

The proposed scene text detection technique has been evaluated on two public available datasets, ICDAR2011 [17], ICDAR2013 [18], and follows the standard evaluation protocol in this field. Both datasets have been widely used as the standard benchmarks for text detection in natural images. In addition, it has been compared with some state-of the-art techniques over the two datasets.

3.1 Data and Evaluation Metric

The datasets used in ICDAR 2011is inherited from the benchmark used in the previous ICDAR competitions, but have undergone extension and modification, since there are some problems with the previous dataset, for example, imprecise bounding boxes and definition of errors. It includes 299 training images and 255 testing images.

The ICDAR 2013dataset is a subset of ICDAR 2011. Several images that duplicated over training and testing sets of the ICDAR 2011 dataset are removed. Meanwhile, a small number of the ground truth annotations are revised. It includes 229 training images, 233 testing images.

The performance of our method is quantitatively measured by precision, recall and F-measure. For the ICDAR2011 and ICDAR2013 datasets, there are three kinds of matching: one-to-one, one-to-many, many-to-one. The evaluation method used in ICDAR 2011 was originally proposed by Wolf et al. [19]. The evaluation protocol for ICDAR 2013 is similar with that of ICDAR 2011, expect for a number of heuristics cues. For more details, please refer to [18].

3.2 Experimental Results

Table 1 Text detection results on ICDAR2011 dataset (%)

Algorithm	Year	Precision	Recall	F-measure
Proposed	2015	80.93	**75.68**	**78.22**
Huang et al. [20]	2014	**88.00**	71.00	78.00
Zamberletti et al. [21]	2014	86.00	70.00	77.00
Yin et al. [22]	2014	86.29	68.26	76.22
Neumann and Matas [23]	2013	85.40	67.50	75.40
Yin et al. [24]	2015	83.77	66.01	73.84
Yao et al. [25]	2014	82.20	65.70	73.00
Kim et al. [26]	2011	82.98	62.47	71.28

Table 2 Text detection results on ICDAR2013 dataset (%)

Algorithm	Year	Precision	Recall	F-measure
Proposed	2015	80.61	**75.16**	77.79
Lu et al. [27]	2015	**89.22**	69.58	**78.19**
Yin et al. [22]	2014	88.47	66.45	75.89
Neumann and Matas [28]	2012	87.51	64.84	74.49
Yin et al. [24]	2015	83.98	65.11	73.35
Shi et al. [29]	2013	84.70	62.85	72.16
I2R NUS FAR [30]	2013	75.00	69.00	72.00
I2R NUS[30]	2013	73.00	66.00	69.00

The performance of the proposed approach on the two datasets is shown in Table 1 and 2. In ICDAR 2011 dataset, our method achieved the precision, recall and F-measure of 80.93%, 75.68% and 78.22%, respectively. In ICDAR 2013 dataset, our method achieved the precision, recall and F-measure of 80.61%, 75.16% and 77.79%, respectively, the winning algorithm in the ICDAR Robust Reading Competition 2013 reports a F-score of 75.89% while our technique obtains 77.79% as shown in Table 2. As the two tables show, the proposed technique obtains similar results for the ICDAR2011 dataset and ICDAR2013 dataset and it outperforms state-of-the-art techniques clearly in recall especially, even if the precision should be improved in the future.

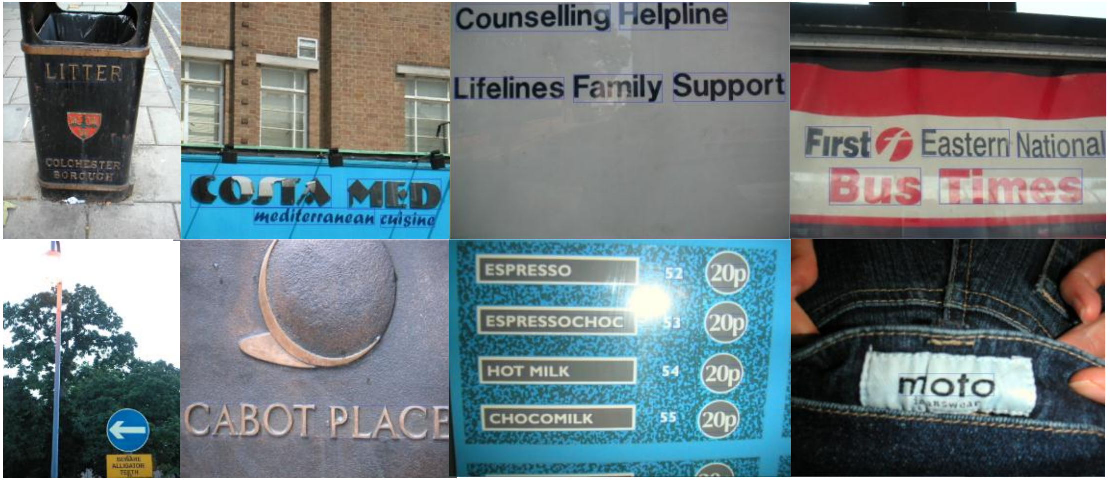

Figure 3. Successful samples

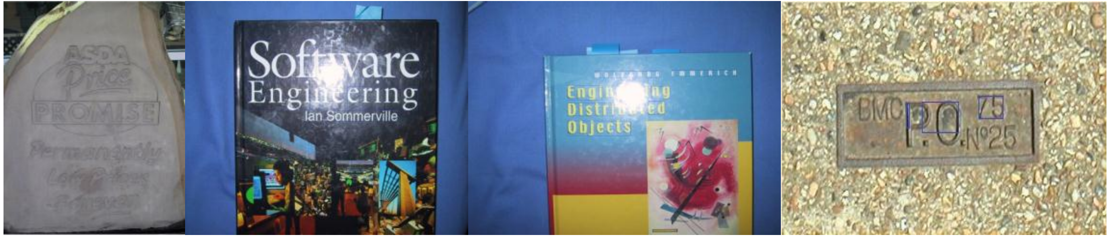

Figure 4. Failed samples

REFERENCES

[1] Sam S. Tsai, Huizhong Chen, David M. Chen, Georg Schroth, Radek Grzeszczuk, and Bernd Girod, "Mobile visual search on printed documents using text and low bit-rate features," ICIP, pp. 2601–2604, (2011)

[2] Trung Quy Phan, Palaiahnakote Shivakumara, and Chew Lim Tan, "Text detection in natural scenes using gradient vector flow-guided symmetry," ICPR, 3296–3299, (2012)

[3] Chucai Yi and Yingli Tian, "Text string detection from natural scenes by structure-based partition and grouping," IEEE Trans. Image Processing, vol. 20, no. 9, 2594–2605, (2011)

[4] Boris Epshtein, Eyal Ofek, and Yonatan Wexler, "Detecting text in natural scenes with stroke width transform," CVPR, 2963–2970, (2010)

[5] Ali Mosleh, Nizar Bouguila, and A. Ben Hamza, "Image text detection using a bandlet-based edge detector and stroke width transform," BMVC, 1–12, (2012)

[6] Trung Quy Phan, Palaiahnakote Shivakumara, and Chew Lim Tan, "Text detection in natural scenes using gradient vector flow-guided symmetry," ICPR, 3296–3299, (2012)

[7] Lukas Neumann and Jiri Matas, "Text localization in real-world images using efficiently pruned exhaustive search," ICDAR, 687–691, (2011)

[8] Lukas Neumann and Jiri Matas, "Real-time scene text localization and recognition," CVPR, 3538–3545, (2012)

[9] Hyung Il Koo and Duck Hoon Kim, "Scene text detection via connected component clustering and non-text filtering," IEEE Trans. Image Processing, vol. 22, no. 6, 2296–2305, (2013)

[10] Sam S. Tsai, Huizhong Chen, David M. Chen, Georg Schroth, Radek Grzeszczuk, and Bernd Girod, "Mobile visual search on printed documents using text and low bit-rate features," ICIP, 2601–2604, (2011)

[11] Yi-Feng Pan, Xinwen Hou, and Cheng-Lin Liu, "A hybrid approach to detect and localize texts in natural scene images," IEEE Trans. Image Processing, vol. 20, no. 3, 800–813, (2011)

[12] Chucai Yi and Yingli Tian, "Localizing text in scene images by boundary clustering, stroke segmentation, and string fragment classification," IEEE Transactions on Image Processing, vol. 21, no. 9, 4256–4268, (2012)

[13] L. Neumann and J. Matas. A method for text localization and recognition in real-world images. ACCV, (2010)

[14] L. Neumann and J. Matas, "Scene textlocalization and recognition with oriented stroke detection," ICCV, (2013)

[15] W. Huang, Z. Lin, J. Yang, and J. Wang. Text localization in natural images using stroke feature transform and text covariance descriptors. ICCV, (2013)

[16] Cheng-Lin Liu, Kazuki Nakashima, Hiroshi Sako, and Hiromichi Fujisawa. Handwritten digit recognition: investigation of normalization and feature extraction techniques. Pattern Recognition, vol.37, no.2004, 265–279, (2004)

[17] A. Shahab, F. Shafait, and A. Dengel. ICDAR 2011 robust reading competition challenge 2: Reading text in scene images. In Proc. of ICDAR, (2011)

[18] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. de las Heras. ICDAR 2013 robust reading competition. ICDAR, (2013)

[19] C. Wolf and J. M. Jolion. Object count/area graphs for the evaluation of object detection and segmentation algorithms. IJDAR, 8(4):280–296, (2006)

[20] W. Huang, Y. Qiao, and X. Tang. Robust scene text detection with convolution neural network induced mser trees. ECCV, 497 – 511, (2014)

[21] A. Zamberletti, L. Noce, and I. Gallo. Text localization based on fast feature pyramids and multi-resolution maximally stable extremal regions. ACCV, 91 – 105, (2014)

[22] X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao. Robust text detection in natural scene images. In IEEE Transactions on Pattern Analysis and Machine Intelligence, 970 – 983, (2014)

[23] L. Neumann and J. Matas. On combining multiple segmentations in scene text recognition. ICDAR, 523 – 527, (2013)

[24] X.-C. Yin, W.-Y. Pei, J. Zhang, and H.-W. Hao. Multiorientation scene text detection with adaptive clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1930 – 1937, (2015)

[25] C. Yao, X. Bai, and W. Liu. A unified framework for multioriented text detection and recognition. IEEE Transactions on Image Processing, pages 4737 – 4749, (2014)

[26] A. Shahab, F. Shafait, and A. Dengel. ICDAR 2011 robust reading competition: Reading text in scene images. ICDAR, pages 1491 – 1496, (2011)

[27] S. Lu, T. Chen, S. Tian, J.-H. Lim, and C.-L. Tan. Scene text extraction based on edges and support vector regression. International Journal on Document Analysis and Recognition, 1 – 11, (2015)

[28] L. Neumann and J. Matas. Real-time scene text localization and recognition. CVPR, 3538 – 3545, (2012)

[29] C. Shi, C. Wang, B. Xiao, Y. Zhang, and S. Gao. Scene text detection using graph model built upon maximally stable extremal regions. Pattern recognition letters, 107 – 116, (2013)

[30] ICDAR 2013 robust reading competition challenge 2 results. http://dag.cvc.uab.es/icdar2013competition, (2014).