Efficient Text Localization in Born-Digital Images by Local Contrast-Based Segmentation

Kai Chen*, Fei Yin*, Amir Hussain† and Cheng-Lin Liu*
*National Laboratory of Pattern Recognition (NLPR), Institute of Automation of Chinese Academy of Sciences,
95 Zhongguancun East Road, Beijing 100190, P.R. China
Email: {kchen, fyin, liucl}@nlpr.ia.ac.cn
†Department of Computing Science and Mathematics, University of Stirling, Stirling FK9 4LA, UK
Email: ahu@cs.stir.ac.uk

***Abstract*—Text localization in born-digital images is usually performed using methods designed for scene text images. Based on the observation that text strokes in born-digital images mostly have complete contours and the pixels on the contours have high contrast compared with the adjacent non-text pixels, we propose a method to extract candidate text components using local contrast. First, the image is segmented into smooth and non-smooth regions. After removing non-text smooth regions, the remaining smooth regions are merged with non-smooth regions to form a candidate text image, which is binarized into high-value and low-value connected components (CCs). The CCs undergo CC filtering, line grouping and line classification to give the text localization result. Experimental results on the born-digital dataset of ICDAR2013 robust reading competition demonstrate the efficiency and superiority of the proposed method.**

***Index Terms*—Text localization, image segmentation, local contrast, connected components grouping.**

I. Introduction

The text elements embedded in born-digital images, prevalent on the Web, carry salient semantic information such as advertisements and security-related information. Born-digital images and scene text images together carry a substantial proportion of information on the Web. Antonacopoulos et al. [1] showed that a large fraction (76%) of text embedded in images cannot be found anywhere else in the web pages. Therefore, extracting text information from born-digital images enhances the semantic relevance of web content for indexing and retrieval. Usually, a text information extraction system consists of three steps: text localization, text segmentation and text recognition. Text localization is critical to the overall system performance and is suffering from variable image background, text color and layout.

Many methods have been proposed for text localization in images, and they roughly fall into two categories: texture-based and connected component (CC)-based.

Texture based methods [2] [3] are based on the observation that text regions in images have distinct textural properties in contrast to non-text regions. Those methods slide a sub-window in multi-scales through all locations of the image using a trained classifier to decide whether the sub-window contains text or not. The exhaustive search makes the computation of texture-based methods costly.

CC based methods first cluster pixels with similar properties (e.g. color, intensity, stroke width, etc) into CCs in the hope that text pixels and non-text pixels are in different CCs. Then text CCs are identified and grouped into text lines. Researchers frequently use color, stroke width transform (SWT) and maximally stable extremal regions (MSERs) to cluster pixels into CCs. Deciding cluster number is the main difficulty for color clustering based methods [4]. SWT based and MSERs based methods are both related to local thresholding. SWT based methods [5] [6] rely heavily on the results of edge detection, which can be seen as local binarization in a small neighborhood. Finding MSERs can be seen as a process to find local binarization results that are stable over a range of thresholds. To reduce the missing of text CCs, MSERs based methods [7] [8] [9] [10] generate tremendous non-text CCs, including many ambiguous ones.

Unlike that many methods have been proposed for scene text detection, few works have been published specifically for born-digital images. Because born-digital images present different characteristics from scene images, it is not necessarily true that methods developed for scene images are appropriate for born-digital images. Text strokes in born-digital images usually have complete contours and pixels on the contours have high contrast compared with the adjacent non-text pixels. This is often not true for text in scene images due to non-ideal camera-capturing environment. Based on this observation, we identify text contour pixels and utilize them to segment an image into text and non-text regions. We then apply binarization to each text region separately to get candidate CCs. Compared to SWT based methods, we have larger regions for binarization. And unlike MSERs based methods, we only need to check a single threshold rather than a range of thresholds. As a result, we have more stable results than SWT based methods, and generate far less non-text CCs than MSERs based methods.

In the next section, we give the details of the proposed method. Section III presents experimental results and Section IV concludes the paper.

II. Method

Our system consists of three stages as shown in Fig. 1. For an image of multi-color text elements and cluttered background, global binarization methods with a single threshold

978-1-4799-1805-8/15/$31.00 ©2015 IEEE

usually cannot separate all text elements from the surroundings. Instead, if we segment the image into non-text and candidate text regions so that each candidate text region contains one type of text, then we can binarize each candidate text region separately to generate candidate text CCs. This is what we do in the first stage and the main contribution of this paper. Then the CCs undergo CC filtering, line grouping and line classification to give the text localization result in the following two stages.

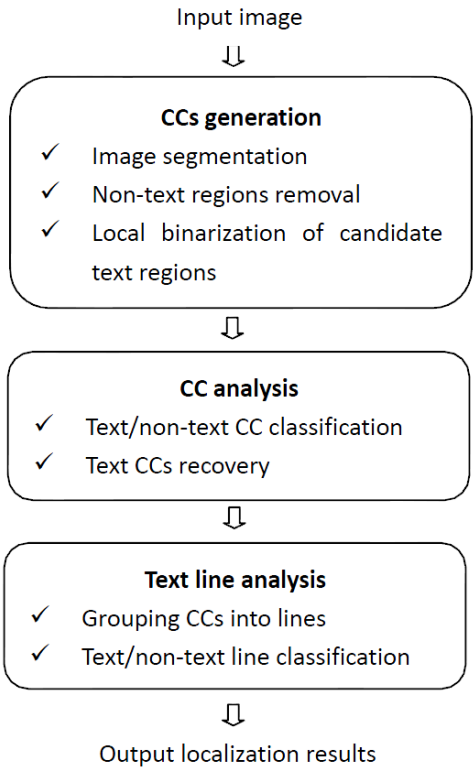

Fig. 1. Block diagram of the proposed algorithm.

A. CCs generation

Text strokes in born-digital images mostly have complete contours, so we can use the contours to detach text pixels from the adjacent non-text pixels. The text pixels on the contours have high contrast compared with the adjacent non-text pixels, and we can identify them based on local contrast thresholding. As illustrated in Fig. 2, we segment an image (Fig. 2(a)) into smooth (Fig. 2(b)) and non-smooth regions (Fig. 2(c)), with pixels of small local contrast constituting smooth regions, and pixels of large local contrast constituting non-smooth regions. The threshold is selected to guarantee that text contour pixels are segmented into non-smooth regions (Fig. 2(c)). The smooth region may also contains text pixels, which are usually in the inner area of strokes. We identify such text smooth regions (Fig. 2(d)), and merge them with non-smooth regions. Then each merged region (Fig. 2(e)) is binarized separately to generate low-value and high-value CCs in image L (Fig.2 (f)) and H (Fig. 2(g)), respectively.

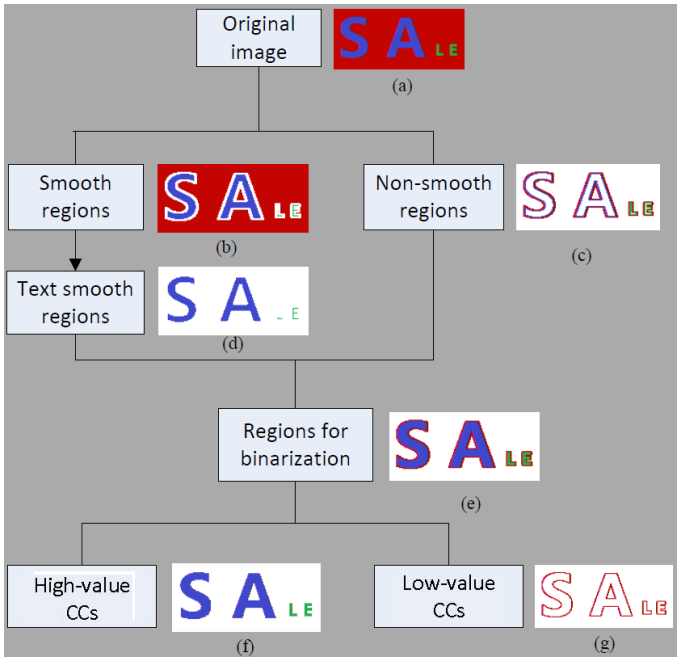

Fig. 2. An illustration for CCs generation. White pixels are used to isolate different regions. (a) An input image. (b) Smooth regions. (c) Non-smooth regions. (d) Text smooth regions selected from b. (e) The merging result of non-smooth regions and text smooth regions. (f) Image L with low-value CCs. (g) Image H with high-valued CCs.

1) Smooth/non-smooth regions segmentation: The gradient magnitude of a pixel can be used to measure the local smoothness. We compute the gradient magnitudes of all pixels and split them by a threshold T selected heuristically. As a result, pixels with magnitudes smaller and larger than T constitute smooth (Fig. 2(b)) and non-smooth regions (Fig. 2(c)), respectively.

For each pixel, we calculate its gradient magnitudes in RGB channels separately using the Sobel operator, and use the largest among the three as the final magnitude.

The threshold is selected to guarantee that text contour pixels are classified into the high-contrast (non-smooth) regions. Assuming that text pixels and neighboring non-text pixels have at least a gap of 15 in a certain channel, which is reasonable for born-digital images, then the gradient magnitudes of those pixels are approximately equal to 60. Therefore, we set T empirically as 60.

2) Text smooth regions selection: We can abandon smooth regions consisting of only non-text pixels without affecting the text localization result. The smooth regions containing text pixels (Fig. 2(d)) are merged with non-smooth regions (Fig. 2(c)) to form a candidate text image (Fig. 2(e)).

When selecting text smooth regions (we treat each region as a CC), we deal with small-sized and large-sized regions differently. Small-sized text regions lose the shapes of the original text (e.g. we cannot read “L” in Fig. 2(d)). Thus we select all small-sized regions to ensure that no small-sized text regions are excluded. On the other hand, we use a text/non-text CC classifier to select large-sized text regions because

TABLE I
11 FEATURES EXTRACTED FROM A CC C FOR TEXT SMOOTH REGION IDENTIFICATION, WITH B, Con, S AND Cs REPRESENTING ITS BOUNDING BOX, CONTOUR, SKELETON AND CONTOUR OF THE SKELETON (FIG. 3 SHOWS AN EXAMPLE), RESPECTIVELY. WE USE THE THINNING ALGORITHM PROPOSED IN [11] TO GET S.

#	Description
1	The Euler number of C
2	The number of pixels in C divided by the area of B
3	The width of B divided by the height of B
4	The area of the convex hull of C divided by the area of B
5	For the CC image, the average number of white-to-black or black-to-white transitions of all rows
6	For the CC image, the average number of white-to-black or black-to-white transitions of all columns
7	The stroke width of C divided by the height of B
8	Stroke width consistency
9	The number of endpoints in S
10	The number of pixels in S divided by the number of pixels in Con
11	The similarity of C and S

they preserve the shapes of the original text (e.g. we can read "SA" in Fig. 2(d)). Heuristically, we identify a region as large-sized if its stroke width is larger than three. We compute the stroke width of a CC as follows. For a CC C, we adopt an idea similar to the one proposed in [12] to assign a value equal to the stroke width to each pixel. We first use distance transform [13] to compute the distance from each pixel to the nearest background pixel, and pixels at the skeleton of C are assigned a value equal to half the stroke width. Then the stroke width information is propagated from the skeleton to the boundary so that every pixel in C has a value representing stroke width. We compute the stroke width of C as the mean of all the values.

For text/non-text CC classification, we use a linear support vector machine (SVM) as the classifier, trained with the 11 features presented in Table I. The features are extracted based on the contour, area, bounding box, skeleton and stroke width of a CC. The first 8 features characterize the properties which have been considered quite often in the problem of text/non-text CC classification. We propose three new features which consider the relationship between the CC and its skeleton. The features #1-7 are calculated straightforwardly. The features #8-11 are elaborated below.

- Feature #8. Via skeletonization and distance transform, we have a value to represent stroke width for each pixel. We measure the stroke width consistency as the standard deviation divided by the mean of all the values.
- Feature #9. We call a pixel an endpoint of a skeleton if it has only one 8-connected neighboring pixel. For example, "F" has three endpoints (Fig. 3(d)). The number of endpoints reflects the number of strokes in a character.
- Feature #10. Because a character and its skeleton share similar structures, we can recognize a character by its skeleton. The more similar a CC is with its skeleton, the closer the value of feature #10 is to 0.5. Therefore, feature #10 reflect this similarity roughly.
- Feature #11. The directional feature of stroke contour has been adopted in character recognition [14], and it is approximately invariant to stroke width variation. Therefore, a text CC should share a similar directional feature of contour with its skeleton. We adopt the method of Liu et al. [14] to compute directional features. A CC image is decomposed into 4 directional subimages by a raster scanning. The codes 0, 1, 2, 3 correspond to horizontal, left-diagonal, vertical and right-diagonal direction, respectively. The feature vector is composed of the histograms of the direction codes, and the similarity of a CC and its skeleton is computed as the cosine similarity between their feature vectors.

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

Fig. 3. An example of a CC and its skeleton. (a) A CC "F". (b) The red rectangle is the bounding box of "F". (c) The contour of "F". (d)The skeleton of "F". (e) The contour of the skeleton of "F".

3) Text regions binarization: After merging text smooth regions with non-smooth regions, most regions in the merged image contain only one type of text (Fig. 2(e)). Therefore, we can apply binarization to each region separately to get text CC candidates. In the sense of classifying each pixel as either foreground or background, binarization is a two-class classification problem. Otsu's binarization algorithm [15] aims to maximize the ratio between inter-class variance and intra-class variance. We apply Otsu's binarization algorithm in RGB channels separately, and choose the channel with the biggest ratio as the result. To facilitate following procedures, we place high-value and low-value CCs in image H and L (Fig. 2(f), (g)), respectively.

There are non-ideal cases when a text region contains not only text pixels and background pixels adjacent to strokes, but also pixels in other parts of the image (e.g. the first column of Fig. 4). However, since text pixels and the adjacent non-text pixels are contrasted, they are very likely to be binarized into different images, and thus form text CCs and non-text CCs separately (e.g. the last two columns of Fig. 4).

B. CC analysis

After binarization in the merged candidate text image as described above, we select text CCs by text/non-text CC classification and recovering some rejected CCs. We adopt the same text/non-text CC classifier as used in Sect.II-A2, and remove the CCs which are labeled as non-text. Then we adopt

Fig. 4. Text pixels and adjacent non-text pixels are separated by binarization. Left: regions isolated by white pixels. Middle: image L with low-value CCs. Right: image H with high-value CCs.

the method of Gonzlez et al. [16] to recover some rejected CCs. The main idea is that a CC should be recovered if it has a neighboring text CC and they meet several heuristic constraints concerning color, stroke width, distance and alignment.

C. Text line analysis

We adopt the method of bai et al. [17] to group CCs into text line candidates. Two neighboring CCs are linked into a pair if they obey several heuristic constraints. Then pairs sharing a common CC are merged sequentially to construct text line candidates until no pairs can be merged. We abandon all lines consisting of less than two CCs and apply text/non-text line classification to the remaining lines. The classifier is a linear SVM with the features in Table II.

Then we use the method of Bai et al. [17] to separate text lines into words according to horizontal distances between consecutive CCs, and combine the results from images H and L. We use arbitrarily oriented minimum bounding boxes (AOMBBs) to represent words. If there are two AOMBBs which have a overlapping area larger than one fourth the smaller one's area, they are replaced by an AOMBB which contains them precisely. We do this sequentially until no more AOMBBs can be merged. At last, an AOMBB localizes a word in the original image.

III. Experimental results

We evaluated the performance of our method on the database of the ICDAR2013 robust reading competition - Challenge 1: Reading Text in Born-Digital Images (Web and Email) [18].

The born-digital dataset comprises images extracted from web pages (news, personal, commercial, social, government, etc) and email messages (spam, newsletters, etc). There are 551 of them with a minimum size of 100 * 100 pixels, out of which 410 and 141 images are used for training and test, respectively. The organizers of the competition maintain a website [19] where the dataset with ground truth can be downloaded after registration. The website also accepts detected results and outputs the evaluation results. The evaluation protocol is described in the report of the competition [18]. It is based on word-level match over the whole test set, taking into account one-to-one, one-to-many and many-to-one matches between detected and ground truth text boxes. We present the results of our method and the methods submitted to the competition in Table III. Brief descriptions of the submitted methods can be found in [18]. The winner was USTB_TexStar, a method based on MSER segmentation. It also won the task of scene text localization.

The results in Table III shows that our proposed method yields superior performance compared to the competition results in ICDAR2013. After CC segmentation, we use conventional techniques for CC filtering, line grouping and word partitioning. The superior performance indicates that the proposed local contrast-based segmentation method is promising.

Fig. 5 shows some examples of successful detection on images in the dataset.

Fig. 6 shows two failure cases in our experiments. For

TABLE II
NINE FEATURES EXTRACTED FROM TEXT LINE CANDIDATE L WITH N CCS $(c_1, c_2, \ldots, c_n)$. FOR A CC, WE DENOTE THE AVERAGE RBG VALUES OF ALL PIXELS AS (r, g, b), THE STROKE WIDTH AS sw AND THE HEIGHT AS h. FEATURE #4, #5 AND #6 ARE FURTHER NORMALIZED TO [0, 1] BY DIVIDING 255.

#	Description
1	The average of the text/non-text CC classification output scores of $c_1, c_2, \ldots, c_n$
2	$\min(sw_1, \ldots, sw_n) / \max(sw_1, \ldots, sw_n)$
3	$\max(sw_1, \ldots, sw_n) - \min(sw_1, \ldots, sw_n)$
4	$\max(r_1, \ldots, r_n) - \min(r_1, \ldots, r_n)$
5	$\max(g_1, \ldots, g_n) - \min(g_1, \ldots, g_n)$
6	$\max(b_1, \ldots, b_n) - \min(b_1, \ldots, b_n)$
7	The height of L divided by the width of L
8	The total horizontal distances between consecutive CCs divided by the width of L
9	The average of regression errors when fitting the centers of $c_1, c_2, \ldots, c_n$ with a straight line using least square error regression divided by the average height of $c_1, c_2, \ldots, c_n$

TABLE III
RANKING OF OUR METHOD AND THE METHODS SUBMITTED TO ICDAR2013 ROBUST READING COMPETITION ON BORN-DIGITAL DATASET.

Method Name	Recall (%)	Precision (%)	F-score(%)
Our method	**85.80**	91.57	**88.59**
USTB TexStar [9]	82.38	**93.83**	87.74
TH-TextLoc	75.85	86.82	80.96
I2R NUS FAR	71.42	84.17	77.27
Baseline	69.21	84.94	76.27
Text Detection [20], [21]	73.18	78.62	75.81
I2R NUS	67.52	85.19	75.34
BDTD CASIA	67.05	78.98	72.53
OTCYMIST [22]	74.85	67.69	71.09
Inkam	52.21	58.12	55.00

Fig. 5. Examples of successful text localization results.

Fig. 6(a), we miss the text because our method cannot handle curved text lines. In Fig. 6(b), we identify a non-text region as text because it has text-like features.

(a) (b)

Fig. 6. Two failure cases of text localization.

IV. Conclusion

We proposed a new CC based method for text localization in born-digital images. By segmenting an image into text and non-text regions based on local contrast, each text region is binarized to generate text CC candidates. The CCs undergo CC filtering, line grouping and line classification to give the final result. Our method has achieved state-of-the-art performance on the born-digital dataset of ICDAR2013 Competition, convincingly demonstrating the effectiveness. We anticipate higher performance if we improve the algorithms for CC classification and line grouping in the future.

Acknowledgment

This work has been supported in part by the National Basic Research Program of China (973 Program) Grant 2012CB316302, the National Natural Science Foundation of China (NSFC) Grant 61411136002 and the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant XDA06040102).

References

[1] A. Antonacopoulos, D.Karatzas, J. Ortiz-Lopez, Accessing textual information embedded in internet images, *Photonics West 2001-Electronic Imaging*, 2000, pp. 198-205.
[2] K.I. Kim, K. Jung, J.H. Kim, Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm, *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(12): 1631-1639, 2003.
[3] Y.-F. Pan, X. Hou, C.-L. Liu, A hybrid approach to detect and localize texts in natural scene images, *IEEE Trans. Image Process.*, 20(3): 800-813, 2011.
[4] J. Park, H. Yoon, G. Lee, Automatic segmentation of natural scene images based on chromatic and achromatic components, *Computer Vision/Computer Graphics Collaboration Techniques*, Springer Heidelberg, 2007, pp. 482-493.
[5] B. Epshtein, E. Ofek, Y. Wexler, Detecting text in natural scenes with stroke width transform, *Int. Conf. on Computer Vision and Pattern Recognition*, 2010, pp. 2963-2970.
[6] W. Huang, Z. Lin, J. Yang, J. Wang, Text localization in natural images using stroke feature transform and text covariance descriptors, *Proc. 14th ICCV*, 2013, pp. 1241-1248.
[7] L. Neumann, K. Matas, Real-time scene text localization and recognition, *Int. Conf. on Computer Vision and Pattern Recognition*, 2012, pp. 3538-3545.
[8] H.I. Koo, D.H. Kim, Scene text detection via connected component clustering and nontext filtering, *IEEE Trans. Image Process.*, 22(6): 2296-2305, 2013.
[9] X.-C. Yin, X. Yin, K. Huang, H.-W. Hao, Robust text detection in natural scene images, IEEE Trans. Pattern Anal. Mach. Intell.
[10] L. Sun, Q. Huo, W. Jia, K. Chen, Robust text detection in natural scene images by generalized colorenhanced contrasting extremal region and neural networks, *Proc. 22nd ICPR*, 2014, pp. 2715-2720.
[11] T.-Y. Zhang, C.Y. Suen, A fast parallel algorithm for thinning digital patterns, *Commun. ACM*, 27(3): 236-239, 1984.
[12] H. Chen, S.S. Tsai, G. Schroth, D.M. Chen, R. Grzeszczuk, B. Girod, Robust text detection in natural images with edge-enhanced maximally stable extremal regions, *Proc. 18th ICIP*, 2011, pp. 2609-2612.
[13] A. Meijster, J.B. Roerdink, W.H. Hesselink, A general algorithm for computing distance transforms in linear time, *Mathematical Morphology and Its Applications to Image and Signal Processing*, Springer US, 2000, pp. 331-340.
[14] C.-L. Liu, Y.-J. Liu, R.-W. Dai, Preprocessing and statistical/structural feature extraction for handwritten numeral recognition, *Progress of Handwriting Recognition*, World Scientific, Singapore, 1997, pp. 161-168.
[15] N. Otsu, A threshold selection method from gray-level histograms, IEEE T-SMC, 1979.
[16] A. Gonzalez, L.M. Bergasa, A text reading algorithm for natural images, *Image Vision Comput.*, 31(3): 255-274, 2013.
[17] B. Bai, F. Yin, C.-L. Liu, Scene text localization using gradient local correlation, *Proc. 12th ICDAR*, 2013, pp. 1380-1384.
[18] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, S.R. Mestre, J. Mas et al., ICDAR 2013 robust reading competition, *Proc. 12th ICDAR*, 2013, pp. 1484-1493.
[19] http://dag.cvc.uab.es/icdar2013competition/
[20] J. Fabrizio, B. Marcotegui, M. Cord, Text detection in street level images, *Pattern Anal. Appl.*, 16(4): 519-533, 2013.
[21] J. Fabrizio, B. Marcotegui, M. Cord, Text segmentation in natural scenes using toggle-mapping, *Proc. 16th ICIP*, 2009, pp. 2373-2376.
[22] D. Kumar, A.G. Ramakrishnan, OTCYMIST: Otsu-Canny minimal spanning tree for born-digital images, *Proc. 10th ICDAR*, 2012, pp. 389-393.