Scene Text Detection Based on Text Probability and Pruning Algorithm

Gang Zhou(✉), Yajun Liu, Fei Shi, and Ying Hu

The Institution of Information Science and Technology, Xinjiang University,
Shengli Road, 14, Ürümqi 830001, China
gangzhou_xju@126.com

Abstract. As the scene text detection and localization is one of the most important steps in text information extraction system, it had been widely utilized in many computer vision tasks. In this paper, we introduce a new method based on the maximally stable extremal regions (MSERs). First, a coarse-to-fine classier estimates the text probability of the ERs. Then, a pruning algorithm is introduced to filter non-text MSERs. Secondly, a hybrid method is performed to cluster connected components (CCs) as candidate text strings. Finally, a fine design classifier decides the text strings. The experimental results show our method gets a state-of-the-art performance on the ICDAR2005 dataset.

Keywords: Scene text detection · Maximum stable extreme regions · Text probability · Pruning algorithm

1 Introduction

As camera and mobile phone become more and more popular, mass images and videos need automatic analysis. Since the text information can be easily understood by computer, the research on the images and videos text information extraction system become an important research in recent years. Scene text detection and localization is the first step in a text information extraction system, and is the most important step in many vision tasks such as web images analysis, mobile phone translation and so on.

The characters in natural scene vary in size, color and font, and should be very hard to distinct from the complex background (see Fig. 1). Such characteristics of scene text detection made it a challenging research. In many comprehensive surveys of scene text detection [1], the methods can be classified into two categories: the region-based methods and the CCs-based methods. Region-based methods extract texture feature vectors by a sliding window strategy in different scales and positions of images. Candidate text regions are decided by rules or classifiers, and then are merged to generate text blocks. Although this kind of methods are robust to the influence of natural light and the image blur, it needs a large amount computation and gets a imprecise location results. In our early work [2], a cascade classifier is introduced to detect local sliding regions by three kinds of features. The results got a very high recall ratio but a very poor precision ratio. In another work [3], a conditional random field model is introduced to combine the local similarity sliding windows as a unity, and improved the precision ratio.

D.-S. Huang et al. (Eds.): ICIC 2016, Part III, LNAI 9773, pp. 726–735, 2016.
DOI: 10.1007/978-3-319-42297-8_67

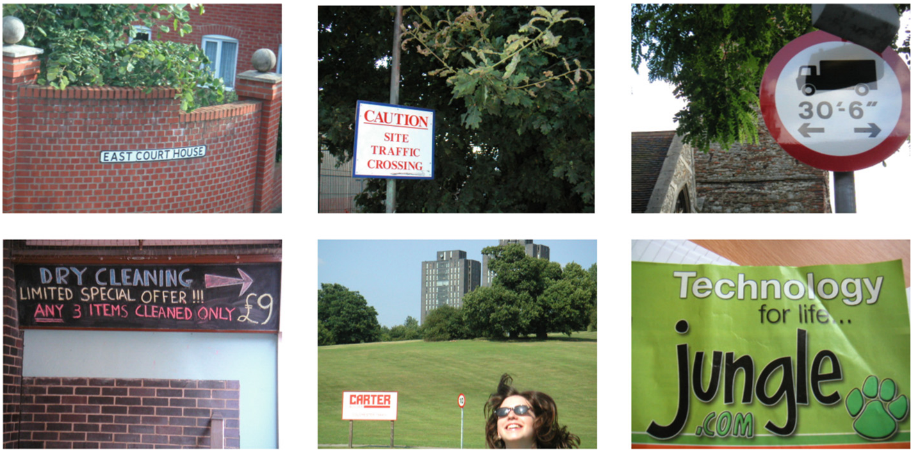


Fig. 1. The scene text image samples

On the other hand, CCs-based methods segment candidate CCs from images, and the non-text CCs are pruned with appearance of unary components. The candidate text CCs are then clustered as CCs unites for the sequent analysis. This approach is attractive because the results can be directly used for character recognition. In our early work [4], the images were over-segmented into super-pixels, and candidate CCs were extracted by combining local contrast and color consistency. The non-text components are then pruned by a hierarchical model consisting of three stages in cascade. In another work [5], a text probability map consisting of the text position and scale information is estimated to segment CCs. To filter out the non-text CCs, a hierarchical model consisting of two classifiers in cascade is utilized to filter.

In this paper, we apply the MSER methods to extract CCs. And then, text probability is estimated by a coarse-to-fine classifier and is utilized to filtered the non-text CCs by a pruning algorithm. The candidate text CCs are clustered into strings which can be determined by a fine designed classifiers. The rest of this paper is organized as follows: (1) related work is in Sect. 2, (2) the detail of this method is introduced in Sect. 3, (3) the experiments are shown in Sect. 4, (4) the conclusions are given in Sect. 5.

2 Related Work

There are two key parts in CCs-based methods, CCs segmentation and analysis. In CCs segmentation parts, the MSER method becomes the most efficient one. This method is introduced in [6] which are widely utilized in scene text detection. In [7], the MSERs are extracted with a fixed delta and is processed with a CCs analysis part and a CCs cluster algorithm. In [8], a more common extremal region (ER) method is introduced. The ERs is extracted not only on grey channel but also in other channel. Then, incremental computable feature sets with a cascade classifier are utilized for filtering the non-text ERs. In [9], the author proposes a novel frame takes advantages of both MSERs and sliding-window based methods. The MSERs operator dramatically reduces the number of windows scanned and enhances detection of the low-quality texts. While the sliding-window with convolutional neural network is applied to correctly separate the connections of multiple characters in components. In [10], a fast and effective pruning algorithm is designed to extract MSERs as character candidates using the strategy of minimizing regularized variations.

In CCs analysis part, the researchers are focused on extraction of distinct features and CCs cluster. A stroke width transform algorithm is introduced in [11]. The stroke width map is then utilized to extract CCs and becomes a powerful feature. In [12], the approach mirrors the move from saliency detection methods to measures of objectiveness. In order to determine the characters, the author develops three novel cues that are tailored for character detection and a Bayesian method for their integration. In [13], both character appearance and structure are combined to generate representative and discriminative text descriptors.

CCs cluster part is utilized for exploring more context information. To efficiently filter out the non-text components, a conditional random field model considering unary component properties and binary contextual component relationships with supervised parameter learning is proposed in [14]. A new text model is constructed to describe text unit which includes two characters and the link connection. For every candidate text unit, they combine character and link energies to compute text unit energy which measures the likelihood that the candidate is a text object.

3 Scene Text Detection Method Introduction

In this paper, we propose a MSER-based method. The text probability is estimated by a coarse-to-fine classifier and is utilized to filter the non-text MSERs with a pruning algorithm. The candidate CCs are then linked by a rule-based algorithm and a classifier based algorithm. The linked CCs are merged into the CCs strings which are then classified as text regions and non-text regions. The flowchart of our method is shown in Fig. 2.

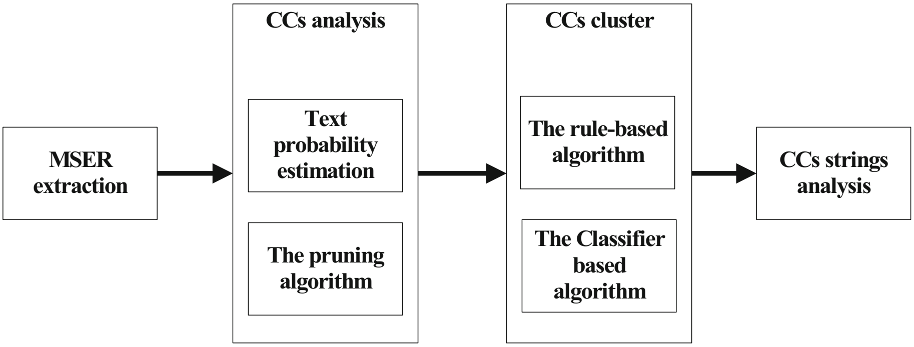


Fig. 2. The flowchart of our method

3.1 CCs Analysis

The MSER algorithm will segment repeated components which are overlap in the same region. The repeat components can be combined as a MSER tree in a parent-children relationship, see Fig. 3(a). To filter the repeat components will help us to find the proper text CCs and reduce the large amount number of non-text CCs.

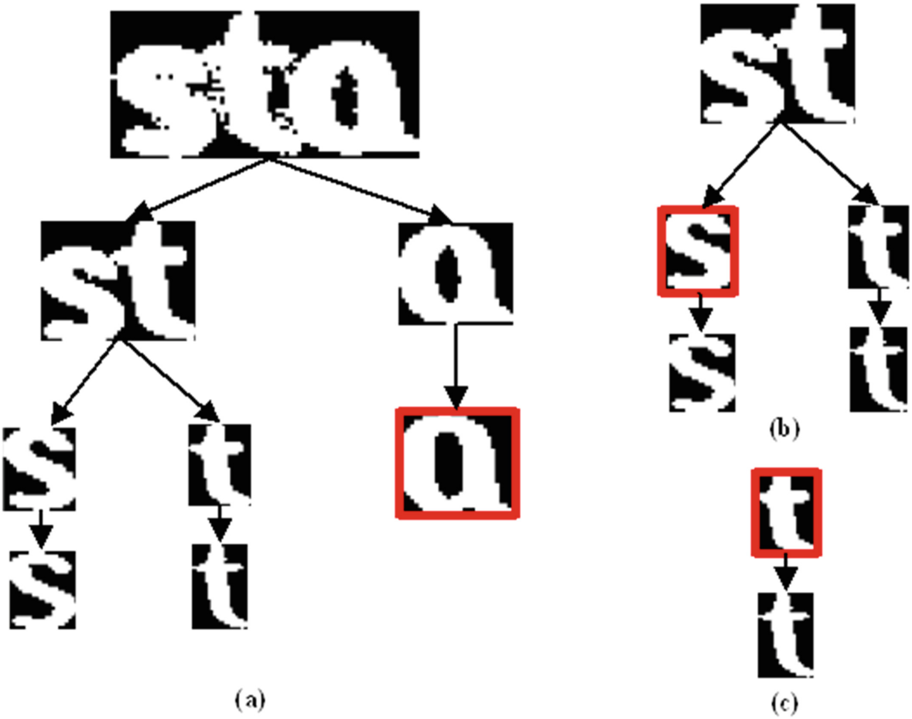


Fig. 3. The MSER tree pruning algorithm. The region with red bounding box get maximum text probability. (a) First pruning. (b) Second pruning. (c) Third pruning.

3.1.1 Text Probability Estimation

The text probability of every MSER will be estimated by a cascade classifier with two layers. In the first layer, 4 features with low computation are given with an Adaboost classifier. In the second layer, 8 features (include 4 features in the first layer) are given with a SVM classifier to estimate the text probability.

The first layer will pruning most of non-text MSERs, and 4 features are the height-width ratio, the occupation ratio, the compactness, and convex hull ratio. The second layer will add 4 more features to estimate the text probability: the boundary gradient, the stroke width occupation ratio, the stroke width difference, and the stroke width-height ratio. All the details of these features can be found in [4, 8].

3.1.2 Pruning Algorithm

In a MSER tree, all the CCs are linked as father nodes or children nodes. We first label all the CCs as '0' means not processed. Then, the CC with the maximum text probability will be selected and labeled as '1' means a text CC. As show in Fig. 3(a), the character 'a' with red bounding box is a candidate text CC. All the ancestor nodes and descendant nodes of this candidate text CC will be labeled as '2' means non-text CCs. In such process, we will get a new MSER tree as shown in Fig. 3(b). Then, we will do the same process in this MSER tree until there is no CC labeled as '0'. We can easily find out that three CCs 'a', 's', and 't' will be labeled as candidate text CCs in three steps, as shown in Fig. 3(a–c).

3.2 CCs Cluster

Since the characters in natural scene always clustered as words or strings, the CCs cluster step can integrate more context information for filtering the non-text CCs. In this paper, a rule-based algorithm is considered to cluster CCs with prior location relationship. Then, a classifier based algorithm filters the link between non-text CC and text CC. The candidate CCs will be clustered with the link relationship.

3.2.1 The Rule-Based Algorithm

The nearby characters always locate in horizontal arrangement in natural scene. The ith CC located in a bounding box whose centre point is (x_i^c, y_i^c), and the top and down axis of this CC is y_i^t and y_i^d. The jth CC will be linked with the ith CC follow two rules, as shown in following.

$$\left| x_i^c - x_j^c \right| < 4 \cdot \max(h_i, h_j) \tag{1}$$

$$\min(y_i^d, y_j^d) - \max(y_i^t, y_j^t) > 0.5 \cdot \min(h_i, h_j) \tag{2}$$

The h_i and h_j are the height of the ith CC and the jth CC. The Eq. 1 gives a nearby relationship, and the Eq. 2 gives a horizontal arrangement relationship. Following these two rules, the CCs will be linked (as shown in Fig. 4(b)).

3.2.2 The Classifier Based Algorithm

The nearby characters always shared similar color, stroke width and height. To eliminate the links between the text CCs and non-text CCs, a SVM classifier is utilized to decide the text CCs links. In the training step, the links between the text CCs are labeled as positive samples and the links between the non-text CCs and text CCs are labeled as negative samples (other links are not labeled). Then, 5 features are extracted to describe the links: the color difference, the stroke difference, the height difference, mean text probability of two CCs, and text probability difference. All the details of these features can be found in our previous work [4].The linked CCs will be clustered as candidate CCs strings.

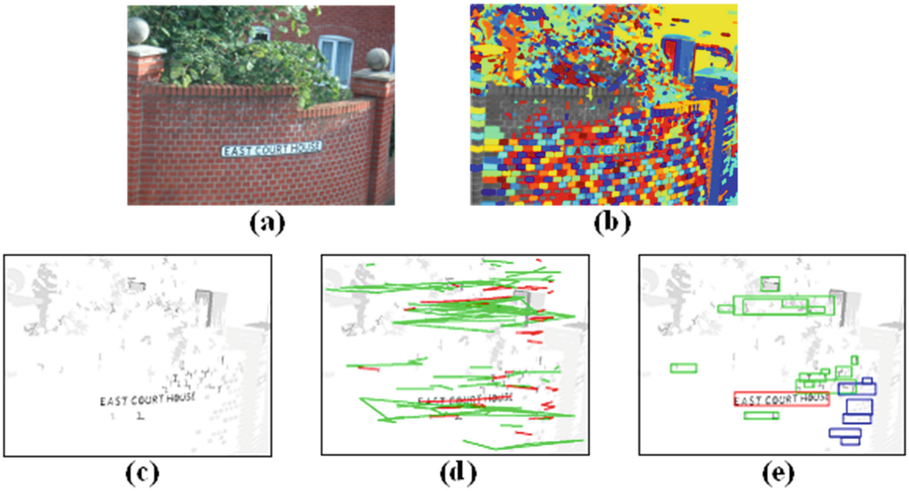


Fig. 4. CCs cluster and CCs strings analysis. (a) The original image. (b) The MSERs. (c) The text probability of the candidate CCs. (d) The CCs cluster results. All the links are produced by the rule based algorithm. The red links are decided by the classifier based algorithm. (e) The red bounding box is the text CCs, the green bounding box is the first kind of non-text CCs strings, and the blue bounding box is the second kind of non-text CCs strings. (Color figure online)

3.3 CCs Strings Analysis

Although the characters can be linked as a text string after CCs cluster step, some of non-text CCs also can be clustered as CCs strings. From the labeling step, we found out that there are two kinds of non-text CCs strings in natural scene. In first kind, some of the non-text CCs (such as leaves) are occasionally clustered as CCs strings. This kind of CCs strings always appears as low text probability, small number of CCs, and so on. In second kind, some of repeat CCs (such as barrels, bricks, windows) are clustered as CCs strings. This kind of CCs strings always appears as same shape, see Fig. 4(c).

From the observation of the labeling step, 5 features are built up for a SVM classifier. As shown in next:

- Mean text probability. In a text CCs strings, the text probability of every CC will be higher than the non-text CCs strings.
- The number of CCs. The number of CCs in a text CCs string will be higher.
- The stroke height ratio. The ratio between the mean stroke of every CCs and the height of the CCs string region can separate the text CCs stings from the repeat CCs strings (the bricks and barrels always appear high ratio).
- The occupation ratio difference. The occupation ratio in every CC of a text string will be higher than the repeat CCs strings.
- The stroke width occupation ratio difference. The stroke width occupation ratio in every CC of a text string will be higher.

As the text strings must be split into words, we use the bounding box distance to measure the distance between words [4].

4 Experimental Results

In experiments, we focus on analysis of every step and the final location results. In Subsect. 4.1, the parameters of the classifiers of three steps (CCs analysis, CCs cluster and CCs strings) are discussed. In Subsect. 4.2, the detection and localization of our method on the ICDAR2005 dataset will be compared with other methods. The performance of our approach is first evaluated on the public ICDAR2005 dataset. The ICDAR2005 dataset includes 258 images in the training set and 251 images in the test set. The text characters in these images include English and numeral characters and are horizontally aligned.

4.1 The Parameters Discussion

In the CCs analysis step, there are two stages: Adaboost classifier and SVM classifier. In Adaboost classifier, 4 features are extracted from the labeled samples. The RealAdaboost algorithm is utilized, and about 1.8 k positive samples and 32 k negative samples are training the parameters. In Fig. 5(a), the ROC curve of the Adaboost classifier obtained by cross-validation. The threshold used in the experiments is 2.2 (recall 95.85 %, precision 65.83 %).

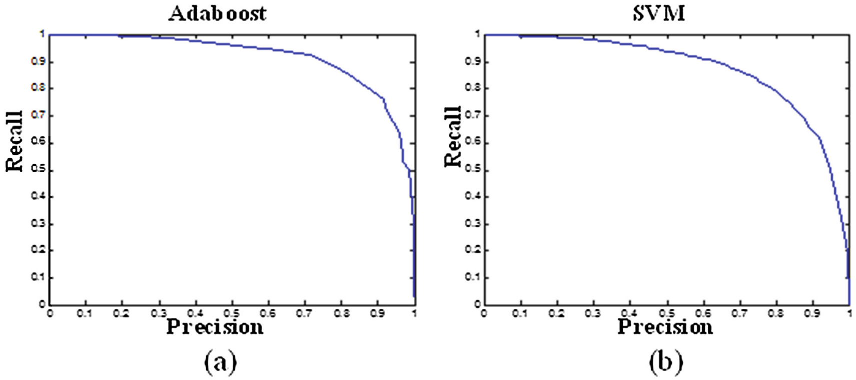


Fig. 5. The performance in CCs analysis step. (a) The Adaboost classifier performance. (b) The SVM classifier performance.

In SVM classifier, 8 features are extracted from 13 k samples (about 1.7 k positive samples and 11.3 k negative samples). In Fig. 5(b), the ROC curve of the SVM classifier obtained by cross-validation. And a logistic correction algorithm is obtained to estimate the text probability. The threshold used in the experiments is 0.05 (recall 94.83 %, precision 50.16 %).

In the CCs cluster step, a SVM classifier is obtained in classifier based algorithm. In the CCs pairs labeling, we found out there are less negative samples (about 120 negative samples and 400 positive samples). When the threshold value is 0, the classifier performance obtained 93 % recall ratio and 97 % precision ratio. And in the CCs strings step, we also use a SVM classifier to analyze the text strings. In this step about 273 positive samples and 322 negative samples are labeled. When the threshold value is 0, we got about 95 % recall ratio and 82 % precision ratio.

4.2 The Localization Results

After we obtain the text localization results, the standard precision-recall terms [15] are obtained to evaluate the word localization result. The word detection result is a set of rectangles designating bounding boxes for detected words. A set of ground truth boxes is provided in the dataset. The match value between two rectangles is defined as the area of the intersection rectangle divided by the area of the minimum containing rectangle. Hence, the best match for a rectangle in a set of rectangles can be defined as the maximum match value.

Our approach got about 69 % recall ratio, 72 % precision ratio and 70 % F-measure performance which is better than our previous work. Comparing with some other public papers, our approach is also better. All the comparison results can be seen in Table 1.

Table 1. The comparision results

Approach	Recall %	Precision %	F-measure %
Proposed methods	69	72	70
Gang et al. [4]	69	70	69
Epshtein et al. [11]	60	73	66
Pan et al. [14]	67	70	69
Gang et al. [5]	67	68	67

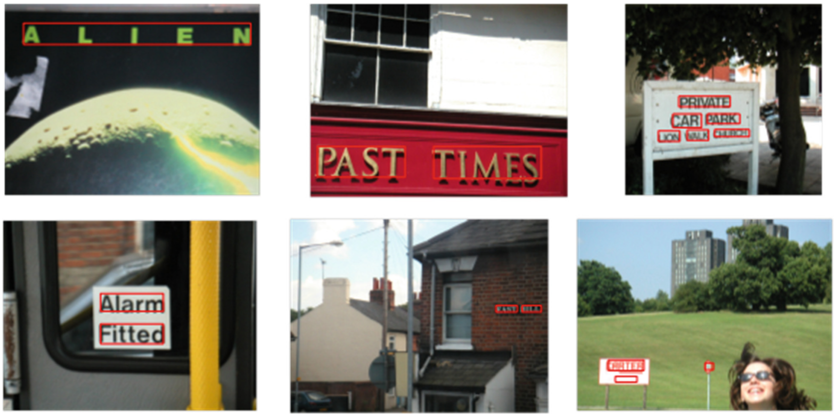
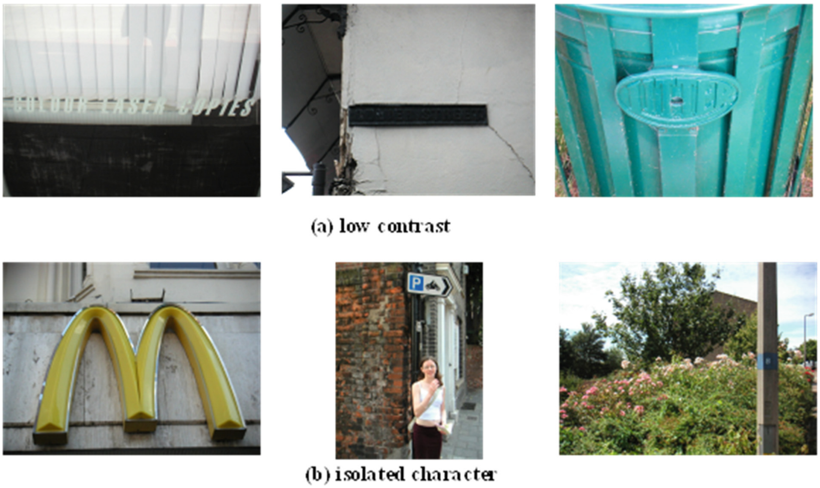


Fig. 6. Some of successful samples

Some of the succeed samples of localization results can be seen in Fig. 6. Our approach can get precise results, and the results can be recognized directly.

Some of the failed samples of localization results can be seen in Fig. 7. Our approach failed on some nature images due to two categories. In first category, some of characters can not be extracted by the MSER algorithm. As the MSER algorithm processed on the intensity channel, some characters in scene do not appear contrast in gray value, see Fig. 7(a). In second category, character is isolated in images. In this situation, the text CC can not be clustered, and can not be properly processed in sequential step, see Fig. 7(b).



(a) low contrast

(b) isolated character

Fig. 7. Some of failed samples

5 Conclusion

In this paper, we propose a scene text detection method based on estimating text probability and pruning algorithm. Three steps including CCs analysis, CCs cluster, and CCs strings analysis are obtained in our approach. In CCs analysis step, a coarse-to-fine classier estimates the text probability of the MSERs. Then, a pruning algorithm based on select maximum text probability is introduced. In CCs cluster step, a rule based algorithm and a classifier based algorithm are proposed to link CCs pairs. In CCs strings analysis step, a SVM classifier decides the text strings. The experimental results discussed the parameters in each step and analyzed the localization results on ICDAR2005 dataset. Our method got better results than other 4 methods. However, some low contrast and isolated character samples are failed by our method which needs to improve in our future work.

Acknowledgements. The authors thank the anonymous reviewers and editor who provided many valuable comments and suggestions. This work is funded by Scientific Research Programs of the Higher Education Institution of XinJiang (grant nos. XJEDU2014S006 and XJEDU2014I004).

References

1. Ye, Q.X., Doermann, D.: Text detection and recognition in imagery: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **37**(7), 1480–1500 (2015)
2. Zhou, G., et al.: Detecting multilingual text in natural scene. In: Proceedings of the 2011 1st International Symposium on Access Spaces (ISAS), pp. 116–120 (2011)
3. Zhou, G., et al.: A new hybrid method to detect text in natural scene. In: 2011 18th IEEE International Conference on Image Processing (ICIP 2011), pp. 2605–2608 (2011)
4. Zhou, G., et al.: Scene text detection method based on the hierarchical model. IET Comput. Vision **9**(4), 500–510 (2015)
5. Zhou, G., Liu, Y.H.: Scene text detection based on probability map and hierarchical model. Opt. Eng. **51**(6), 067204-1–067204-9 (2012)
6. Matas, J., et al.: Robust wide-baseline stereo from maximally stable extremal regions. Image Vis. Comput. **22**(10), 761–767 (2004)
7. Koo, H.I., Kim, D.H.: Scene text detection via connected component clustering and nontext filtering. IEEE Trans. Image Process. **22**(6), 2296–2305 (2013)
8. Neumann, L., Matas, J.: Real-time scene text localization and recognition, In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3538–3545. IEEE, New York (2012)
9. Huang, W., Qiao, Y., Tang, X.: Robust scene text detection with convolution neural network induced MSER trees. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part IV. LNCS, vol. 8692, pp. 497–511. Springer, Heidelberg (2014)
10. Yin, X.C., et al.: Multi-orientation scene text detection with adaptive clustering. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1930–1937 (2015)
11. Epshtein, B., et al.: Detecting text in natural scenes with stroke width transform. In: IEEE Conference on Computer Vision and Pattern Recognition 2010, pp. 2963–2970. IEEE Computer Society, Los Alamitos (2010)
12. Li, Y., et al.: Characterness: an indicator of text in the wild. IEEE Trans. Image Process. **23** (4), 1666–1677 (2014)
13. Yi, C.C., Tian, Y.L.: Text extraction from scene images by character appearance and structure modeling. Comput. Vis. Image Underst. **117**(2), 182–194 (2013)
14. Pan, Y.F., Hou, X.W., Liu, C.L.: A hybrid approach to detect and localize texts in natural scene images. IEEE Trans. Image Process. **20**(3), 800–813 (2011)
15. Lucas, S.M., et al.: ICDAR 2003 robust reading competitions: entries, results, and future directions. Int. J. Doc. Anal. Recogn. **7**(2–3), 105–122 (2005)