Cloud of Line Distribution and Random Forest Based Text Detection from Natural/Video Scene Images

Wenhai Wang[1], Yirui Wu[2], Palaiahnakote Shivakumara[3], and Tong Lu[1(✉)]

[1] National Key Lab for Novel Software Technology, Nanjing University, Nanjing, China
wangwenhai362@163.com, lutong@nju.edu.cn
[2] College of Computer and Information, Hohai University, Nanjing, China
wuyirui@hhu.edu.cn
[3] Department of Computer System and Information Technology, University of Malaya, Kuala Lumpur, Malaysia
shiva@um.edu.my

Abstract. Text detection in natural and video scene images is still considered to be challenging due to unpredictable nature of scene texts. This paper presents a new method based on Cloud of Line Distribution (COLD) and Random Forest Classifier for text detection in both natural and video images. The proposed method extracts unique shapes of text components by studying the relationship between dominant points such as straight or cursive over contours of text components, which is called COLD in polar domain. We consider edge components as text candidates if the edge components in Canny and Sobel of an input image share the COLD property. For each text candidate, we further study its COLD distribution at component level to extract statistical features and angle oriented features. Next, these features are fed to a random forest classifier to eliminate false text candidates, which results representatives. We then perform grouping using representatives to form text lines based on the distances between edge components in the edge image. The statistical and angle orientated features are finally extracted at word level for eliminating false positives, which results in text detection. The proposed method is tested on standard database, namely, SVT, ICDAR 2015 scene, ICDAR2013 scene and video databases, to show its effectiveness and usefulness compared with the existing methods.

Keywords: COLD · Random forest · Text detection in natural scene image
Text detection in video image

1 Introduction

Text detection in natural scene video images is important for many real time applications such as driving vehicle without pilots and robotic applications to identify place or person in office automatically [1, 2]. It is true that a real dataset usually consists of

K. Schoeffmann et al. (Eds.): MMM 2018, Part II, LNCS 10705, pp. 48–60, 2018.
https://doi.org/10.1007/978-3-319-73600-6_5

mixed data, namely, images and videos captured by a variety of devices. For example, on Facebook and WhatsApp, we can see both images and videos. This makes the text detection problem more challenging. Many methods have been developed in the past years for solving the issues of text detection in natural scene and video images in multimedia community [1, 2]. However, these methods usually focus on one type of dataset and may not perform well for different datasets as they report inconsistent results. In addition, the images uploaded in Facebook and Instagram usually suffer from contrast variation and poor quality. Achieving better detection rate for such images and videos is challenging. These factors motivated us to propose a new method for text detection in both natural and video images in this work, which should also be robust to multi-type texts, scripts and distortions to some extent.

2 Related Work

The methods of text detection can be classified broadly into the method which focus on text in natural scene images, the methods which focus on text in video images and the methods which focus both on text in natural and video images. In this work, we review the above three categories to highlight the gap between the state-of-the-art and the problem mentioned in the Introduction section.

Feng et al. [3] proposed scene text detection based on multi-scale SWT and edge filtering, which explores edge density and stroke width information. The main focus of this method is to detect texts from natural scene images, so it expects high contrast images for achieving better results. Pei et al. [4] proposed multi-orientation scene text detection with multi-information fusion, which explores a convolutional neural network classifier and an Adaboost classifier for text detection from natural scene images. In addition, poor results are reported for blurred or low contrast images. Wu et al. [5] proposed natural scene text detection by multi-scale adaptive color clustering and non-text filtering. Zheng et al. [6] proposed a cascaded method for text detection in natural scene images. This method uses two classifiers at character level for text detection. However, the focus of the method is only natural scene images. In addition, the method does not work well for multi-oriented text images. Yin et al. [7] proposed robust text detection in natural scene images based on MSER and classifiers. Neumann and Mattas [8] proposed real time scene text localization and recognition, which explores MSER concept. Li et al. [9] proposed a method for text detection in natural scene images, which explores a Bayesian classifier and the Markov Random Field model. Mosleh et al. [10] proposed an automatic inpainting scheme for video text detection and removal, which explores stroke width transform for text detection in video. Mittal et al. [11] proposed rotation and script independent text detection from video frames using sub-pixel mapping. This method explores super resolution concept for enhancing low contrast text information, and then using descriptor to extract features. Despite addressing the challenges of multi-oriented, low contrast and multi-script, the method reports inconsistent results on different types of data. This is due to variation in contrast and background for different types of datasets.

In the same way, there are methods for detecting texts in both natural scene and video images. For example, Wu et al. [12] proposed contour restoration of text components for recognition in video and scene images, which restores the loss of edges in video for text detection and recognition. Shivakumara et al. [13] proposed a fractals based multi-oriented text detection system for recognition in mobile video images. Shivakumara et al. [14] also proposed a new multi-modal approach to bib number/text detection and recognition in Marathon images. These approaches are good for images which do not suffer much from illumination effect and blur. Recently, deep learning methods are also explored for text detection in natural scene images. For instance, Huang et al. [15] proposed robust scene text detection with convolutional neural networks induced MSER trees. Jaderberg et al. [16] proposed reading texts in the wild with convolutional neural networks. It is true that the use of deep learning still has inherent limitations, such as labeling a large number of samples especially for non-texts, for which there is no boundary or limit to choose good samples for training. In addition, setting optimal parameters for different datasets is not so easy. Furthermore, it requires good infrastructure to execute deep learning.

It is noted from the above review that most methods focus on particular data type but not both natural and video type images which affected by poor quality and contrast variations. Hence, there is a need to develop robust method which fills the gap.

Therefore, in this work, we propose a new method based on Cloud of Line Distribution (COLD) [17], which extracts shapes of text components according to linearity and non-linearity of contours in polar domain. The proposed method has two major contributions. Firstly, since the proposed method considers the distribution of text components in polar domain, the proposed method is robust to distortion to some extent and is independent of scripts, orientations and text types. Secondly, the computation cost for extracting COLD and constructing related classifiers is quite low, which makes the proposed method highly effective for text detection tasks. Therefore, the proposed method is appropriate to deploy in embedded systems, which often lack powerful computation resources. These advantages of the proposed method result in achieving better results for both natural and video images with less computation cost.

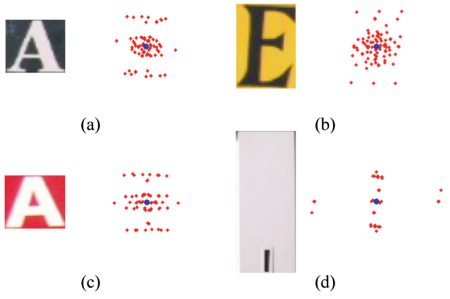

Fig. 1. The COLD for different kinds of components, where (a), (b) and (c) show text components and their COLD, and (d) gives a non-text component and its COLD, respectively.

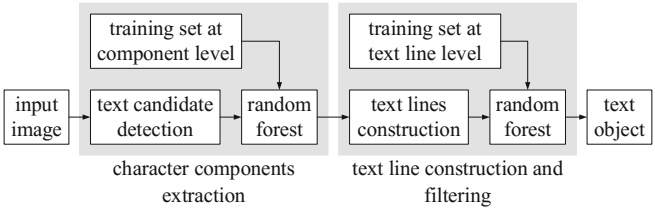

Fig. 2. The overview of the proposed method.

3 The Proposed Method

It is true that text components in images of different types have different characteristics on shape, contrast, stroke information, or uniform spacing. Well-designed features are expected to be robust for the same character even there are rotations, scaling and distortions, meanwhile distinguishable for different characters. We show several examples of the proposed COLD for different kinds of components in Fig. 1. It can be observed from Fig. 1 that the same character (Two "A") tends to own similar distributions, while the two different characters ("A" and "E") have different distributions. Motivated by the COLD for different kinds of components, we apply COLD for text detection.

Figure 2 gives the overview of the proposed method. The details of the proposed approach are as follows. Firstly, we propose a method based on COLD for detecting text candidates. Next, we study the COLD of different k at component level to extract statistical features and angle oriented features, where k refers to the parameter which denotes the difference of array indexes between two dominant points on the dominant points array. Figure 3(b) and (c) show the process of defining dominant point pairs with $k = 1$ and $k = 2$, respectively. Figure 3(d) and (e) show the corresponding result distributions with log-polar space of (b) and (c). We then feed these features to a random forest classifier to get character components. After that, we use the character components to form text lines by aligning the spacing between character components. Finally, we extract statistical and angle orientated features at text line level and feed these features to a random forest classifier to eliminate false text lines.

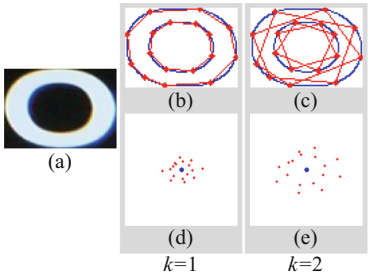

Fig. 3. Illustration of the process of the dominate point pair detection and the corresponding result distributions respectively, where (a) is the example text region, (b) and (c) describe the process of defining dominant point pairs with $k = 1$ and $k = 2$, respectively, (d) and (e) show the corresponding result distributions with log-polar space of (b) and (c).

3.1 Text Candidate Detection

For an input image, we extract text candidates by the method based on COLD. We propose to use the method in [17] to compute COLD. It is observed from Fig. 4 that the COLD of Canny and Sobel images of character components are almost the same, and the COLD of Canny and Sobel images of background region are very different. Therefore, we generate text candidates according to the following steps:

(i) Get Canny and Sobel images for the input image, compute the COLD ($k = 1, 2, 3$) of Canny image, which is named as C_C, and similarly compute the COLD of Sobel image, which is named C_S.
(ii) Find common points between C_C and C_S. In other word, compute the intersection set C_I of C_C and C_S.
(iii) Compute the corresponding dominant points $\{d_i\}$ of C_I, and restore the edge image corresponding to the dominant points $\{d_i\}$ and Canny image of the input image.
(iv) Get edge components of the restored edge image. The edge components can be regards as text candidates of the input image.

The results of these four steps are illustrated in Fig. 5, where (a) gives an input image, (b) and (c) show its Canny and Sobel images, (d) and (e) respectively give the COLD of (b) and (c), (f) shows the common information between (d) and (e), (g) is the restored image, and (h) gives edge components.

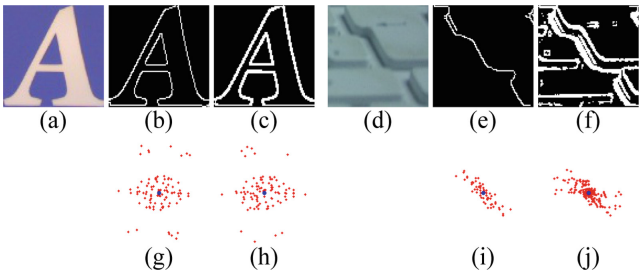

Fig. 4. The result of COLD for Canny and Sobel images of different component. (a) Is character component, (b) and (c) are Canny and Sobel images for (a), (g) and (h) are COLD of (b) and (c). Where (d) is background region, (e) and (f) are Canny and Sobel images for (d), (i) and (j) are COLD of (e) and (f).

3.2 Representatives Detection for Text Line Construction

In this subsection, we aim to extract statistical features and angle oriented features from COLD at component level. For each text candidate, we calculate its COLD distribution of different k, which is named as C_k ($k = 1, 2, 3, \ldots, 10$), where k refers to the parameter that denotes the distance on the dominant sequence. We perform experiments to select $k = 10$. For each C_k, we calculate distances between every point pairs, which are named as $d_1, d_2, \ldots, d_m$. Then we extract features $\{\mu_k, \sigma_k, \theta_k\}$ as in Eq. 1:

$$\begin{cases} \mu_k = \dfrac{\sum_{i=1}^{m} d_i}{m} \\ \sigma_k = \sqrt{\frac{1}{n}\sum_{i=1}^{m} (d_i - \mu_k)^2} \\ \theta_k = PCA(C_k) \end{cases} \tag{1}$$

where μ_k and σ_k refer to the mean and standard deviation value of distances, respectively, θ_k represents the angle of C_k, function $PCA()$ represents the operation of Principal Component Analysis (PCA). Here we use PCA to calculate the angle of C_k.

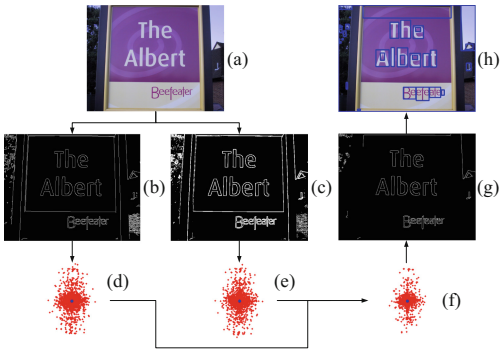

Fig. 5. The overview of text candidates detection based on COLD, where (a) is the input image, (b) and (c) are Canny and Sobel images of the input image, (d) and (e) are COLD of (b) and (c), (f) is common information between (d) and (e), (g) is the restored edge image corresponding to the common information in COLD, (h) gives edge components of the restored edge image.

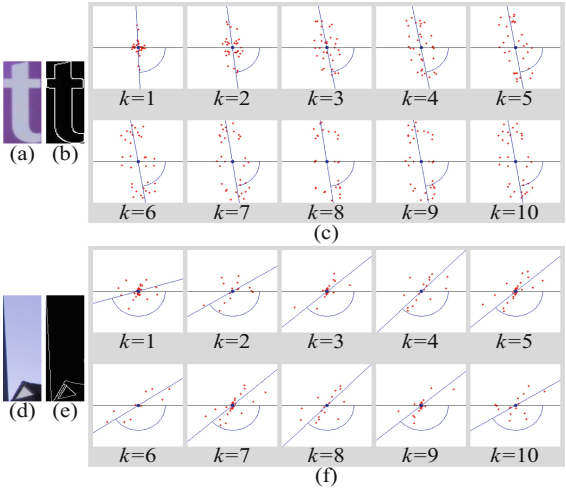

Fig. 6. The feature construction at component level, where (a) is a text candidate, (b) is restored edge image of (a), (c) is COLD for (b) of different $k(k = 1, 2, 3, \ldots, 10)$, (d) is a false text candidate, (e) is restored edge image of (d), (f) is COLD for (e) of different $k(k = 1, 2, 3, \ldots, 10)$.

Next, we combine the mean, standard deviation and angle of each C_k to combine them into a 30 dimensional feature vector. Feature construction at component level for text and non-text examples are shown in Fig. 6, where we can find that with the increasing of k, the θ of text components changes less than non-text components, and the distribution of text components will be more scattered and then tend towards stability, while that of non-text components cannot be stabilized. At last, we apply the feature vector at component level in a random forest classifier to assign labels for each candidate. Example results after eliminating false text candidates are shown in Fig. 7.

3.3 Text Lines Construction

This subsection presents text line extraction by grouping text candidates given by the previous step. Let the text candidate set be S, which provides coarse locations of texts. To draw bounding boxes for extracted text lines, the proposed method groups text candidates which share the common shape properties such as height, scale and orientation. Specifically, the proposed method extracts multi-oriented texts with the conditions in Eq. 2:

(a) (b)

Fig. 7. The result after eliminating false text candidates, where (a) is the result of text candidates, and (b) is the result after eliminating false text candidates.

$$f_\rho(p,q) = 1, if \begin{cases} 2/3 \le |H_p/H_q| \le 3/2 \\ |\theta_p - \theta_q| \le \pi/8 \\ |f_d(p,q) - (W_p + W_q)/2| \le H_p + H_q \end{cases} \tag{2}$$

where H, W and θ represent height, width and orientation of a text region, respectively, and $f_d(p,q)$ refers to the distance between the center of p and that of q. Note that the parameters in Eq. 2 are determined experimentally. If a text candidate satisfies this property, the method groups it. In this way, the proposed method groups text candidates into text lines. Then we extract the feature vector from COLD of every text line as the previous step. Feature construction at text line level is shown in Fig. 8. At last, we feed the feature vector at text line level to a random forest classifier to eliminate false text lines. The effect of these steps is shown in Fig. 9.

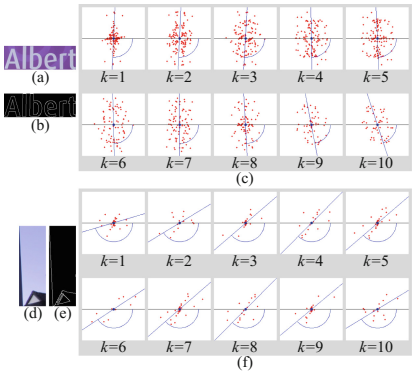

Fig. 8. The feature construction at text line level, where (a) is a text line, (b) is restored edge image of (a), (c) is COLD distribution of different k (k = 1, 2, 3, …, 10), (d) is a false text line, (e) is restored edge image of (d), and (f) is COLD distribution of different k (k = 1, 2, 3, …, 10).

Fig. 9. The result of text line, where (a) is the result after eliminating false text candidates, (b) is the result of text lines after grouping text candidates by heuristic method, (c) is the result after eliminating false text lines.

4 Experiments

To evaluate the proposed method, we consider four benchmark databases, namely, ICDAR 2013 scene, ICDAR 2015 scene, ICDAR 2013 video, Street View Text (SVT). To calculate measures for text detection in both natural scene images and video frames, we follow the standard evaluation scheme as in the ICDAR robust competition [18]. According to the instructions given in [18], we calculate Recall, Precision and F-measure to evaluate the performance of the proposed method. Moreover, we record time-cost per image on a PC (2.9 GHz 2-core CPU, 8G RAM, no GPU device involved) to compare the computation cost. In order to show the effectiveness of the proposed method, we use available codes of Yin et al. [7], Neumann and Mattas [8] and Li et al. [9], which use the MSER concept for text detection. In the same way, we implement Wu et al.'s method [12], which explores character shape restoration for text detection in both natural and video images, and Huang et al.'s method [15] which uses the concepts of MSER and convolutional neural networks for text detection. On top of this, we also reported the results listed in [19] for the database of ICDAR 2015.

Note that the training set for random forest at component level is extracted from several benchmark datasets, namely, ICDAR 2013 scene/video, ICDAR 2015 scene, and SVT. There are totally 109545 components (25980 text components and 83565 non-text components) for training. The training dataset for random forest at text line level is extracted from these datasets as well. There are totally 9045 text lines (8242 true text lines and 803 false text lines) for training.

We train the random forest classifier with the parameters as follows. The number of the trees in the forest is set as 10. The function to measure the quality of a split is Gini impurity. The number of features to consider when looking for the best split is set as 8. The maximum depth of the tree is set as 6, and the minimum number of samples required to split an internal node is set as 2. Finally, the minimum number of samples required to be at a leaf node is set as 1.

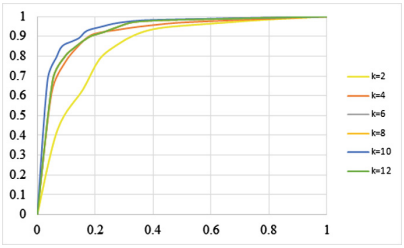

Fig. 10. The ROC curve of different k. The blue curve is the ROC curve of k = 10. (Color figure online)

Table 1. Performance of text detection on ICDAR 2013 scene.

Method	Precision	Recall	F-measure	Time-cost (s)
Proposed	**0.87**	0.66	**0.75**	1.42
Yin et al. [7]	0.84	0.65	0.73	1.73
Neumann and Matas [8]	0.74	0.63	0.68	**0.40**
Wu et al. [12]	0.78	0.62	0.69	1.93
Huang et al. [15]	0.86	**0.67**	**0.75**	3.23

Table 2. Performance of text detection on SVT.

Method	Precision	Recall	F-measure	Time-cost (s)
Proposed	**0.77**	0.64	**0.70**	1.47
Yin et al. [7]	0.41	0.66	0.51	1.89
Neumann and Matas [8]	0.65	0.63	0.64	**0.61**
Wu et al. [12]	**0.77**	0.65	**0.70**	2.05
Huang et al. [15]	0.74	**0.67**	**0.70**	4.03

Table 3. Performance of text detection on ICDAR 2015 scene.

Method	Precision	Recall	F-measure
Proposed	0.70	0.38	0.49
CNN Pro.	0.35	0.34	0.35
Deep2Text	0.50	0.32	0.39
HUST	0.44	0.38	0.41
AJOU	0.47	**0.47**	0.47
NJU-Text	0.70	0.36	0.47
StradVision1	0.53	0.46	**0.50**
StradVision2	**0.77**	0.37	**0.50**

4.1 Experiments on Feature of Different *k*

We choose 100 images randomly from all the databases. For these 100 images, we calculate ROC curves of the random forest classifier, which is trained on different feature vectors of different k $(k = 2, 4, 6, 8, 10, 12)$. It is observed from Fig. 10 that the random forest classifier performs the best when $k = 10$.

4.2 Experiments on Natural Scene Images

Quantitative results on natural scene images of the proposed and the existing methods are reported in Tables 1, 2 and 3 for the ICDAR 2013 scene data, SVT and ICDAR 2015 scene data, respectively. Table 1 shows that the proposed method is the best at F-measure and Precision, while Recall and time-cost is the second best for ICDAR 2013 scene database. This shows the effectiveness of the proposed method upon traditional text detection methods and some of the CNN-based methods. Neumann and Matas method [8] is designed for real-time text detection, which achieves the best result (0.4 s) in computation cost. However, it achieves a much lower F-score (0.48) than the proposed method (0.75). Huang et al.'s method [15] is as good as the proposed method in F-score. However, running a CNN-based architecture could be time-consuming without powerful GPU device, which could be proved by its largest time-cost (2.83 s). Similarly, Table 2 shows the proposed method scores the best results for SVT dataset with reasonable computation cost. For ICDAR 2015 scene, the existing methods offer less information about both codes and cost-time. We simply copy the result of the existing methods to make a comparison. It is observed from Table 3 that the proposed method scores the third best results at Recall, while Precision and F-measure are the second best for ICDAR 2015 scene database compared with the existing methods. Considering that most of the existing methods list in Table 2 are CNN-based methods, we can notice the high computation cost without the support of GPU device, which are often true in embedded systems. Detection examples of the proposed method on ICDAR 2013 scene, ICDAR 2015 scene and SVT are shown in Fig. 11. We can notice the proposed method is robust to distortion to some extent and is independent of scripts, orientations and text types since we use the distribution of text components in polar domain for features.

Fig. 11. Detection examples of the proposed method on ICDAR 2013 scene, ICDAR 2015 scene and SVT.

4.3 Experiments on Video Images

Quantitative results on video images of the proposed and the existing methods are reported in Table 4 for ICDAR 2013 video data. Since video images are greatly affected by low-contrast, multi orientation and non-uniform illumination compared to scene images, we can notice the drop in F-measure for the proposed method and Yin et al. [7] in Table 4. The proposed method is still the best at Precision and F-measure, while Recall is the second best for the ICDAR 2013 video data. We can notice the proposed method achieves the best detection result with the reasonable time-cost. This also shows the consistence of the proposed method when dealing with datasets of various features. Detection examples of the proposed method on ICDAR 2013 video are shown in Fig. 12.

Table 4. Performance of text detection on ICDAR 2013 video

Method	Precision	Recall	F-measure	Time-cost (s)
Proposed	**0.65**	0.63	**0.64**	1.56
Mosleh et al. [10]	0.50	0.49	0.49	**0.81**
Yin et al. [7]	0.64	0.57	0.60	1.76
Li et al. [9]	0.46	**0.70**	0.56	1.98

Fig. 12. Detection examples of the proposed method on ICDAR 2013 video.

5 Conclusion

In this paper, we have proposed a new method for text detection in natural scene, video images. We propose COLD for detecting text candidates. For text candidates, the proposed method first extracts statistical features and angle oriented features of text candidates. To eliminate false text candidates, we then feed feature vectors at component level to a random forest. Next, we perform grouping useful representatives to form text lines based on the distance between edge components in the edge image. The statistical and angle orientated features are finally extracted at text line level for eliminating false positives. We have tested the proposed method on different databases to show the effectiveness of the proposed method. Our future work includes the further improvement of the proposed method and the discovery on distribution patterns of more text types.

Acknowledgements. This work was supported by the Natural Science Foundation of China under Grant 61672273, Grant 61272218, and Grant 61321491, the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant BK20160021, the Science Foundation of JiangSu under Grant BK20170892, the Fundamental Research Funds for the Central Universities under Grant 2013/B16020141, and the open Project of the National Key Lab for Novel Software Technology in NJU under Grant KFKT2017B05.

References

1. Ye, Q., Doermann, D.: Text detection and recognition in imagery: a survey. IEEE Trans. PAMI 1480–1500 (2015)
2. Yin, X.C., Zuo, Z.Y., Tian, S., Liu, C.L.: Text detection, tracking and recognition in video: a comprehensive survey. IEEE Trans. Image Process. 2752–2773 (2016)
3. Feng, Y., Song, Y., Zhang, Y.: Scene text detection based on multi-scale SWT and edge filtering. In: Proceedings of ICPR, pp. 634–639 (2016)
4. Pei, W.Y., Yang, C., Kau, L.J., Yin, X.C.: Multi-orientation scene text detection with multi-information fusion. In: Proceedings of ICPR, pp. 646–651 (2016)
5. Wu, H., Zou, B., Zhao, Y.Q., Chen, Z., Zhu, C., Guo, J.: Natural scene text detection by multi-scale adaptive color clustering and non-text filtering. Neurocomputing 1011–1025 (2016)
6. Zheng, Y., Li, Q., Ju, J., Hu, H., Li, G., Zhang, S.: A cascaded method for text detection in natural scene images. Neurocomputing 1–9 (2017)
7. Yin, X., Yin, X., Huang, K., Hao, H.: Robust text detection in natural scene images. IEEE Trans. PAMI **36**(5), 970–983 (2014)
8. Neumann, L., Matas, J.: Real-time scene text localization and recognition. In: Proceedings of CVPR, pp. 3538–3545 (2012)
9. Li, Y., Jia, W., Shen, C., Hengel, A.V.D.: Characterness: an indicator of text in the wild. IEEE Trans. IP **23**(4), 1666–1677 (2014)
10. Mosleh, A., Bouguila, N., Hamza, A.B.: Automatic inpainting scheme for video text detection and removal. IEEE Trans. IP **22**(11), 4460–4472 (2013)
11. Mittal, A., Roy, P.P., Singh, P., Raman, B.: Rotation and script independent text detection from video using sub pixel mapping. Vis. Commun. Image Represent. (2017, to appear)

12. Wu, Y., Shivakumara, P., Lu, T., Lim Tan, C., Blumenstein, M., Kumar, G.H.: Contour restoration of text components for recognition in video/scene images. IEEE Trans. IP **25**(12), 5622–5634 (2016)
13. Shivakumara, P., Wu, L., Lu, T., Tan, C.L.: Fractals based multi-oriented text detection system for recognition in mobile video images. Pattern Recogn. 158–174 (2017)
14. Shivakumara, P., Raghavendra, R., Qin, L., Raja, K.B., Lu, T., Pal, U.: A new multi-modal approach to bib number/text detection and recognition in Marathon images. Pattern Recogn. 479–491 (2017)
15. 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. LNCS, vol. 8692, pp. 497–511. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10593-2_33
16. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. IJCV **116**(1), 1–20 (2016)
17. He, S., Schomaker, L.: Beyond OCR: multi-faceted understanding of handwritten document characteristics. Pattern Recogn. 321–333 (2017)
18. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., Boorda, L.G.I., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De las Heras, L.P.: ICDAR 2013 robust reading competition. In: Proceedings of ICDAR, pp. 1115–1124 (2013)
19. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanow, A., Iwamura, M., Matas, J., Neumann, L., Chandrsekhar, V.R.: ICDAR 2015 competition on robust reading. In: Proceedings of ICDAR, pp. 1156–1160 (2015)