A review of methods for text detection in imagery of natural scenes

Boyu Zhang

School of Computer Science, Northeast Electric Power University, Jilin, Jilin province, 132012, China
byzhang025@gmail.com

Abstract. Text detection in natural scenes plays a key role in recognition and understanding of text in computer vision, while providing the basic possibilities for combining with Natural Language Processing (NLP). The ability of detection for small text and adjacent text is weak although the text detection in natural scenes has evolved from the horizontal text detection initially to the arbitrary shape text detection. Meanwhile, the environment of natural scenes is complex and changeable, which is a huge challenge for text detection. In this paper, we discuss and compare the horizontal and multi-directional detection methods based on target detection, as well as the curved and arbitrary shape detection methods based on the instance segmentation and semantic segmentation. The algorithm ideas are described and their advantages and disadvantages are analysed. Moreover, we found two research routes of curved text and arbitrary shape text detection known as Top-Down and Bottom-Up. According to the evaluations, PMTD [20] performs best in multi-directional text detection, while SPCNet [16] wins in the curve and arbitrary shape text detection.

1. Introduction

Reading text in natural scenes is one of the most popular fields in computer vision across the world in recent years. It extracts information from text in the image and has a wide range of application scenarios, such as photo translation, scenario understanding, identification of road signs for autonomous car and so on. Reading text in natural scenes is split into two processes of Text Detection and Text Recognition, however, it is not exactly similar with Optical Character Recognition (OCR). OCR is an input technology for the computer that converts printed materials (such as newspapers, books, and manuscripts, etc.) scanned by optical input methods into image and converts them into computer-recognized characters. It uses character segmentation for recognition, without finding the location of text.

As a necessary pre-processing of text recognition, the goal of text detection in natural scenes is to localize all areas of text that can be discerned by human vision. In the reading of scene texts, therefore, scholars mostly focus on the latter. The research based on deep learning in scene text detection has been conducted widely for the past few years. Based on different shapes of detected bounding box, previous methods for text detection can be divided into three stages, Horizontal/Multi-directional text detection, Curved text detection and Arbitrary-shaped text detection. In recent years, there were many well-performance algorithms announced internationally. However, many of them were aimed at horizontal and multi-directional detection initially. Some of the algorithms were modified from object detection, for instance, Ma et al. [1] and Tian et al. [2] propose algorithms based on Faster R-CNN [3] while Liao et al. [4] and Shi et al. [5] propose algorithms modify from Single Shot MultiBox Detector.
(SSD) [6]. However, what challenges we will face in text detection in natural scenes in the future are more complex, like multi-direction text, curved text or even arbitrary-shape text. That because of most text areas in natural scenes are irregular, and they are more likely to be the shape shown in Figure 1.

![Figure 1. Examples of irregular (multi-directional, curved and nonstandard shape etc.) texts in natural scene.](image)

In order to confront the challenge in the condition of curve, some algorithms like TextSnake [7] were emerged later. And these algorithms were more complex in process of dealing with curved bounding box. Moreover, since the circumstance of the research in arbitrary-shape detection is not much satisfied, there was no definite boundary between curved-text and arbitrary-shape text detection. That is not only because of algorithm complexity, scenic diversity and computational difficulty, but also due to the necessity in arbitrary-shape bounding box for text recognize.

In this paper, we are going to discuss characteristics and defects of some state-of-art and most representative algorithms in the three categories mentioned above. In the second section, we are going to discuss horizontal and multi-directional text detection. In the third section, we will talk about several algorithms in curved text detection. And in the fourth section, arbitrary-shape text detection will be mentioned in detail. After that, experimental comparison will be demonstrated in fifth section. And in the last section, we will discuss the advantages and disadvantages among these algorithms and look forward to future development.

2. Object Detection Idea for Text Detector

Like object detection, text detection algorithms in early period only has the ability of drawing horizontal bounding box, therefore, it only detects horizontal text. They use CNN to find the text area and then segment or use the polygon for the regression of text area. Most of them are based on VGG-16 [8] model. Inspire by Faster R-CNN and LSTM, Tian et al. [2] put forward a method for horizontal text detection. As shown in Figure 2, this algorithm applies bi-directional LSTM [9] to generate final box, using high-density small sliding windows (i.e. connected proposals).

![Figure 2. CTPN.](image)

To solve the poor match problem for default boxes of dense on the horizontal direction while sparse vertically, Liao et al. [4] design a vertical sinking that maintains same aspect ratios and a vertical offset (Figure 3). This detector also altered the size of filter kernels and included “long” default boxes that have large aspect ratios in order to fit the shape of the text area during feature extraction.

However, due to the angle and position of the shooting, the common text shapes in natural scenes are rotated rather than horizontal. Therefore, Liao et al. [10] propose another detector called
TextBoxes++ which have capacity to detect multi-directional text on the basis of TextBoxes. The network outputs regression information instead of the coordinates of the bounding box, and the ground truth regression is achieved by calculating the coordinates of four points (or two points and height). As we could see in Figure 4, regression from the green imaginary line to the yellow full line in TextBoxes++. The green full line is the ground truth minimum circumscribed rectangle, and the black imaginary line box is not matched to the true value of the default box.

![Figure 3. Vertical Sinking in TextBoxes.](image1)

![Figure 4. Boundary box regression in TextBoxes++.](image2)

![Figure 5. Links in SegLink.](image3)

Like CTPN, Shi et al. [5] propose a method to generate a bounding box through Linking Segments. It generates segment information by calculating the default box information and feature map, connecting by Links. Moreover, Links are divided into Within-Layer Link and Cross-Layer Link (Figure 5). The former connects adjacent segments and distinguishes whether they belong to the same instance box, while the latter solves the problem in merge caused by the different size of segments (even if the same location, the size may be inconsistent) outputting from different size of the feature map.

Inspire by the link design in SegLink, Deng et al. [11] detect Multi-directional Text by performing two predictions, text/non-text pixel prediction and join prediction (Figure 6). The eight directions of eight pixels around a single pixel are separately predicted. The connection between two pixels is marked as positive when they are within the same instance. Then the bounding box will be extracted from the feature map by the method in OpenCV after the instance is segmented.

![Figure 6. Eight different directions for connection prediction in PixelLink.](image4)

Since multiple intermediate stages affect the final model effect and the processing speed is slow, Zhou et al. [12] propose a two-stage end-to-end text detection pipeline based on DenseBox and UnitBox. The bounding box prediction process uses U-shape to gradually merge all the feature maps in different sizes (Figure 7), and output the distance from the pixel position to the four boundaries of the rectangle as well as the rotation angle θ.
Inspire by the work of Zhou et al. [13] a new idea (Figure 8) based on SSD is carried by Liao et al. [14] which replaces the common convolution with Rotation-Sensitive Regression (RSR) and uses the Rotary Filter Active Rotating Filters (ARF) for multi-directional convolution to detect multi-directional text.

3. Top-Down Method Based on Instance Segmentation

Curved text is what we often see in natural scenes. Conventional CNN can perform well in feature extraction and boundary box regression in polygons rather than curved text. The detection of curved text is divided into top-down and bottom-up methods. Most top-down methods are based on Mask R-CNN [15]. Moreover, Semantic segmentation methods are commonly used to tackle the problem of detecting curved text. An algorithm of top-down method is proposed by Xie et al. [16]. The feature map of each stage extracted by FPN (Figure 9 (a)) passed to the Text-Context Module (TCM). And the convolved segmentation map in TCM is multiplied to the original image as an attention (Figure 9 (d), blue). Then the feature map (green) is obtained. The detected features and the segmented features are then fused. However, Mask R-CNN will use the classification score of the text box as the final score during the detection process, which will result in the low proportion of the oblique text or curved text in horizontal boxes. To solve this problem, the author introduces the Re-Score Mechanism (Figure 10). The instance segmentation result is projected onto the semantic segmentation graph and output the instance score. The final score is then obtained by calculation.

Moreover, Lyu et al. [17] achieve the detection towards curved text by semantic segmentation of characters (Figure 11). Resnet-50 is used as the skeleton of Feature Pyramid Networks (FPN) [18]. Then text proposals are generated via Roi Align produced by Region Proposal Network (RPN). Fast R-CNN [19] is used for text box regression, and Mask branch is used to implement instance segmentation of text regions and semantic segmentation of single letters.
There is another method proposed by Liu et al. [20] based on Mask R-CNN. The following problems exist in the baseline. Shape information is lost when classifying pixel by pixel. Meanwhile, the transformation of the quadrilateral ground truth to the pixel level ground truth will result in inaccurate data. Furthermore, if the bounding box detected first is inaccurate, the subsequent segmentation result will be inaccurate neither. In order to solve the problems mentioned above, Liu et al. build a new pipeline (Figure 12) called Pyramid Mask Text Detector (PMTD), which predicts a soft text mask for each region of text and then applies plane clustering algorithm to convert the soft mask to the pyramid mask. However, only the results of bounding box using polygon are shown in the paper. Theoretically, the ability to detect curved text can be achieved after modifying the propose method of bounding box.

4. Bottom-Up Method Based on Semantic Segmentation

Bottom-Up is another way for text detection. It detects the characters initially and connects them to form a text line, then the detected text box is proposed. Nowadays, most text detector is transformed into semantic segmentation to find the text centerline by pixel-by-pixel classification, and then the text outline is formed in different ways. Since it is difficult to segment examples with very close distances using the segmentation-based method, Li et al. [21] propose a Progressive Scale Expansion algorithm (PSENet) to solve this problem. Inspired by FPN, they link low-level feature maps with advanced feature maps. After F coding, the result in fusion is projected into N branches, and then the progressive expansion algorithm is used to extend the minimum scale of $S_1$ to $S_n$ to extend the complete shape and obtain the result $R$ (Figure 13).
Long et al. [7] give a new idea of using a series of blue dashed circular discs to deduct the yellow solid text area (Figure 14 (a)). VGG-16 based network is used for feature extraction. After the image is input, the predicted information of the seven feature maps is obtained by Fully Convolutional Networks (FCN) and FPN (Figure 14 (b)). Striding is used to compute the result of the instance segmentation to get the text region skeleton. Then the skeleton will be combined with the predicted radius and the final text region obtained. And it implements the detection of arbitrary shape text. However, the processing speed is slow since the post-processing steps are cumbersome.

5. Evaluations

5.1. Datasets

ICDAR 2015 is proposed in challenge 4 of the ICDAR2015 Robust Reading Competition [22]. This challenge is an accompanying scene text image taken with Google glasses, regardless of positioning, image quality and viewing angle. Therefore, it has a large change in text direction, scale and resolution, which is more difficult than the previous ICDAR challenge. The dataset contains 1,000 training images and 500 test images, and the annotation are provided by a text file for eight sets of position information and one character to show where and what the text is.

ICDAR 2013 is proposed in Challenge 2 of the ICDAR 2013 Robust Reading Competition [23]. It is mainly containing of images of horizontal text, where the text is slightly angled in a portion of image. This data set has been widely used in past text detector, which including 229 training images and 233 test images.

Total-Text is proposed by [24], and different from other data sets. Besides the multi-directional text images, it also contains curved text images that rarely appear in other datasets. The dataset is divided into 1,255 training images and 300 test images. All images are annotated with polygon and word level transcription.

5.2. Platforms

We use the open source codes of the papers and the trained models provided by the authors for the evaluations. Most of the methods are implemented by Caffe (CTPN, TextBoxes, TextBoxes++, RRD and Mask TextSpotter), others are implemented by Tensorflow (SegLink, PixelLink, EAST and SPCNet) or PyTorch (PSENet and TextSnake). The compiling languages are Python (version of 2.7 and 3.6) and C/C++. We unified datasets and experimental platforms, and use the evaluation scripts for test result for unified measurement. All the evaluations are conducted on the platform with Intel Xeon E5 CPU, 64GB memory and NVIDIA P100 GPU. All the parameter settings and input image scaling are same as paper.

5.3. Results and discussions

The evaluations use metrics of Precision, Recall and F-measure. We use the ICDAR 2015 Dataset to experiment for the algorithms. The comparison results are shown in the Table 1. Within the dataset, PMTD implements the best Recall and the best F-measure. Mask TextSpotter achieves the best
Precision. There are only two algorithms achieving more than 90% of Precision, PMTD and Mask TextSpotter. In the second place, PSENet and its proximity to SPCNet surpasses Mask TextSpotter on both Recall and F-measure. Thus, PMTD performs best on this dataset according to Table 2. CTPN performs the worst on this dataset since it only has the ability to detect horizontal text.

Table 1. Results on ICDAR2015*

| Method                   | Precision | Recall | F-measure |
|--------------------------|-----------|--------|-----------|
| CTPN [2]                 | 74.2      | 51.6   | 60.9      |
| TextBoxes++ [10]         | 87.8      | 78.5   | 82.9      |
| SegLink [5]              | 73.1      | 76.8   | 75.0      |
| PixelLink [11] +VGG16 2s | 85.5      | 82.0   | 83.7      |
| PixelLink [11] +VGG16 4s | 82.9      | 81.7   | 82.3      |
| EAST [12] +PVANET2x MS   | 83.3      | 78.3   | 80.7      |
| EAST [12] +PVANET2x      | 83.6      | 73.5   | 78.2      |
| EAST [12] +VGG16         | 80.5      | 72.8   | 76.4      |
| RRD [14]                 | 85.6      | 79.0   | 82.2      |
| RRD [14] MS              | 88.0      | 80.0   | 83.8      |
| SPCNet [16]              | 88.7      | 85.8   | 87.2      |
| PSENet [21]              | 89.3      | 85.2   | 87.2      |
| Mask TextSpotter [17]    | 91.6      | 81.0   | 86.0      |
| PMTD [20]                | 91.3      | 87.4   | 89.3      |
| TextSnake [7]            | 84.9      | 80.4   | 82.6      |

* “2s” stands for conv2*2; “4s” stands for conv4*4; “MS” stands for Multi-Scale; and “PVANET2x” stands for a modified version of PVANET [25], with 2x numbers of channels.

Table 2. Comparison results on ICDAR 2015

Similarly, we conducted a comparative evaluation on the ICDAR 2013 Dataset, and the results are shown in Table 3. SPCNet has achieved the best Recall and F-measure, and Mask TextSpotter has achieved the best Precision. Thus, SPCNet performs best on this dataset according to Table 4. In addition, same algorithm works better on the ICDAR 2013 dataset than the ICDAR 2015 dataset. It can be seen that the difficulty of horizontal text fitting is less difficult than multi-directional text fitting.
Table 3. Result on ICDAR 2013

| Method                | Recall | Precision | F-measure |
|-----------------------|--------|-----------|-----------|
| CTPN [2]              | 83.0   | 93.0      | 87.7      |
| TextBoxes [4]         | 74.0   | 88.0      | 81.0      |
| TextBoxes [4]MS       | 83.0   | 89.0      | 86.0      |
| TextBoxes++ [10]      | 74.0   | 88.0      | 81.0      |
| TextBoxes++ [10] MS   | 86.0   | 92.0      | 89.0      |
| SegLink [5]           | 83.0   | 87.7      | 85.3      |
| PixelLink [11] 2s     | 83.6   | 86.4      | 84.5      |
| PixelLink [11] 4s     | 82.3   | 84.4      | 83.3      |
| PixelLink [11] 2s MS  | 87.5   | 88.6      | 88.1      |
| PixelLink [11] 4s MS  | 86.5   | 88.6      | 87.5      |
| EAST [12]             | 82.7   | 92.6      | 87.4      |
| RRD [14]              | 75.0   | 88.0      | 81.0      |
| RRD [14] MS           | 86.0   | 92.0      | 89.0      |
| SPCNet [16]           | 90.5   | 93.8      | 92.1      |
| Mask TextSpotter [17] | 88.6   | 95.0      | 91.7      |
| TextSnake [7]         | 74.5   | 82.7      | 78.4      |

Table 4. Comparison results on ICDAR 2013

Furthermore, in order to understand the fit of the algorithm on the curved text, we also conducted another comparative evaluation on Total-Text Dataset (Table 5). SPCNet gets all the best score (Table 6), but there is a wide difference between the experimental data of SegLink and SPCNet. It has not happened on other data sets. That is because SegLink is design for multi-directional text, however, this dataset has more curved text, which is difficult for SegLink to fit accurately.
Table 5. Result on Total-Text

| Method             | Precision | Recall | F-measure |
|--------------------|-----------|--------|-----------|
| SegLink [5]        | 30.3      | 23.8   | 26.7      |
| EAST [12]          | 50        | 36.2   | 42        |
| SPCNet [16]        | 83        | 82.8   | 82.9      |
| Mask TextSpotter [17] | 69       | 55     | 61.3      |
| TextSnake [7]      | 82.7      | 74.5   | 78.4      |

Table 6. Comparison results on Total-Text

In summary, the best algorithm (among the algorithm mentioned in this paper) for multi-directional text detection is PMTD (Table 2), and the best algorithm for curve and arbitrary shape text detection is SPCNet (Table 4 and 6). When using the horizontal detection algorithm to detect multi-directional and curved text, there will be a large number of non-text areas in the bounding box, which is difficult to achieve a good fit. Thus, the variety of text areas brings difficulties and challenges to text detection. This will also be the direction for the development of future text detection.

6. Conclusion
In this paper, we introduced some advanced text detection algorithms and discussed their advantages and disadvantages. They have multi-directional, curved and arbitrary shape text detection capabilities. The scene text detection gradually forms a stronger generalization ability from the level detection initially to the multi-directional detection, even curve and arbitrary shape detection. Other methods have also achieved great experimental results. For example, the Conditional Spatial Expansion (CSE) based on the seed extension mechanism for the curved text detection is proposed by Liu et al. [26]. A new idea for detecting the four corners of the bounding box is proposed by Lyu et al. [27]. Liu et al. [28] propose the Markov Clustering Network (MCN) for scene text detection of arbitrary size and direction. Moreover, Xue et al. [29] and Wu et al. [30] propose a three-category classification (text/non-text/border) method based on boundary learning. Also the character level text localization based on weak supervision is proposed by Hu et al. [31] and Tian et al. [32]. These all indicate the importance of the scene text detection and the diversification of ideas, but also represent the difficulty of detection. It is foreseeable in the future that more and more text detection algorithms in research will toward to curved or arbitrary shape text detection, and the ability of detection in adjacent text and small text will be strengthened. Also, more and more end-to-end recognition system for text of arbitrary shapes will be developed based on these well-performed methods of text detection.
References

[1] Ma, J., Shao, W., Ye, H., Wang, L., Wang, H., Zheng, Y., Xue, X. (2018). Arbitrary-oriented scene text detection via rotation proposals. IEEE Transactions on Multimedia, 20(11): 3111-3122.

[2] Tian, Z., Huang, W., He, T., He, P., Qiao, Y. (2016) Detecting text in natural image with connectionist text proposal network. In: European conference on computer vision. Amsterdam. pp. 56-72.

[3] Ren, S., He, K., Girshick, R., Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. Montreal. pp. 91-99.

[4] Liao, M., Shi, B., Bai, X., Wang, X., Liu, W. (2017). Textboxes: A fast text detector with a single deep neural network. In Thirty-First AAAI Conference on Artificial Intelligence. San Francisco.

[5] Shi, B., Bai, X., Belongie, S. (2017). Detecting oriented text in natural images by linking segments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii. pp. 2550-2558.

[6] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C. (2016). Ssd: Single shot multibox detector. In: European conference on computer vision. Amsterdam. pp. 21-37.

[7] Long, S., Ruan, J., Zhang, W., He, X., Wu, W., Yao, C. (2018). Textsnake: A flexible representation for detecting text of arbitrary shapes. In: Proceedings of the European Conference on Computer Vision. Munich. pp. 20-36.

[8] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[9] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8): 1735-1780.

[10] Liao, M., Shi, B., Bai, X. (2018). Textboxes++: A single-shot oriented scene text detector. IEEE transactions on image processing, 27(8): 3676-3690.

[11] Deng, D., Liu, H., Li, X., Cai, D. (2018). Pixellink: Detecting scene text via instance segmentation. In: Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans.

[12] Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J. (2017). EAST: an efficient and accurate scene text detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii. pp. 5551-5560.

[13] Zhou, Y., Ye, Q., Qiu, Q., Jiao, J. (2017). Oriented response networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii. pp. 519-528.

[14] Liao, M., Zhu, Z., Shi, B., Xia, G. S., Bai, X. (2018). Rotation-sensitive regression for oriented scene text detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City. pp. 5909-5918.

[15] He, K., Gkioxari, G., Dollár, P., Girshick, R. (2017). Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. Venice. pp. 2961-2969.

[16] Xie, E., Zang, Y., Shao, S., Yu, G., Yao, C., Li, G. (2019). Scene text detection with supervised pyramid context network. In: Proceedings of the AAAI Conference on Artificial Intelligence. New York. pp. 9038-9045.

[17] Lyu, P., Liao, M., Yao, C., Wu, W., Bai, X. (2018). Mask textsspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In: Proceedings of the European Conference on Computer Vision. Munich. pp. 67-83.

[18] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S. (2017). Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Hawaii. pp. 2117-2125.

[19] Girshick, R. (2015). Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. Santiago. pp. 1440-1448.
[20] Liu, J., Liu, X., Sheng, J., Liang, D., Li, X., Liu, Q. (2019). Pyramid Mask Text Detector. arXiv preprint arXiv:1903.11800.

[21] Li, X., Wang, W., Hou, W., Liu, R. Z., Lu, T., Yang, J. (2018). Shape robust text detection with progressive scale expansion network. arXiv preprint arXiv:1806.02559.

[22] Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., ... Shafait, F. (2015). ICDAR 2015 competition on robust reading. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR). Nancy. pp. 1156-1160.

[23] Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L. G., Mestre, S. R., ... De Las Heras, L. P. (2013, August). ICDAR 2013 robust reading competition. In: 2013 12th International Conference on Document Analysis and Recognition. Washington, DC. pp. 1484-1493.

[24] Ch'ng, C. K., Chan, C. S. (2017, November). Total-text: A comprehensive dataset for scene text detection and recognition. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). Kyoto. pp. 935-942.

[25] Kim, K. H., Hong, S., Roh, B., Cheon, Y., Park, M. (2016). Pvanet: Deep but lightweight neural networks for real-time object detection. arXiv preprint arXiv:1608.08021.

[26] Liu, Z., Lin, G., Yang, S., Liu, F., Lin, W., Goh, W. L. (2019). Towards robust curve text detection with conditional spatial expansion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA. pp. 7269-7278.

[27] Lyu, P., Yao, C., Wu, W., Yan, S., Bai, X. (2018). Multi-oriented scene text detection via corner localization and region segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City. pp. 7553-7563.

[28] Liu, Z., Lin, G., Yang, S., Feng, J., Lin, W., Goh, W. L. (2018). Learning markov clustering networks for scene text detection. arXiv preprint arXiv:1805.08365.

[29] Xue, C., Lu, S., Zhan, F. (2018). Accurate scene text detection through border semantics awareness and bootstrapping. In: Proceedings of the European Conference on Computer Vision. Munich. pp. 355-372.

[30] Wu, Y., Natarajan, P. (2017). Self-organized text detection with minimal post-processing via border learning. In: Proceedings of the IEEE International Conference on Computer Vision. Venice. pp. 5000-5009.

[31] Hu, H., Zhang, C., Luo, Y., Wang, Y., Han, J., Ding, E. (2017). Wordsup: Exploiting word annotations for character based text detection. In: Proceedings of the IEEE International Conference on Computer Vision. Venice. pp. 4940-4949.

[32] Tian, S., Lu, S., Li, C. (2017). Wetext: Scene text detection under weak supervision. In: Proceedings of the IEEE International Conference on Computer Vision. Venice. pp. 1492-1500.