MS TextSpotter: An Intelligent Instance Segmentation Scheme for Semantic Scene Text Recognition in Asian Social Networks*

Yikai Zhong[0009−0001−4578−3649], Heng Zhang[0000−0001−9027−3261] Yanli Liu(✉)[0000−0002−2691−034X], Qiang Qian[0009−0001−4578−3649], and Junwen Wang

School of Electronic Information, Shanghai Dianji University, Shanghai 201306, China
*liuyl@sdju.edu.cn

Abstract. Text detection and recognition in natural scenes is an imimportant task in computer vision. However, most of the texts in natural scenes are curved, and the text background is complex and diverse. In recent years, the text detection and text recognition models proposed have inherent defects, especially in applying many false positives, which usually leads to a decline in text detection and text recognition accuracy. To solve this problem, we propose a text detection and recognition model based on instance segmentation: MS TextSpotter (Mask Scoring TextSpotter), which is based on end-to-end training. First, we design a neural network based on Mask R-CNN. The neural network can achieve accurate text detection and recognition through semantic segmentation, especially for multi-directional and curved text in natural scenes. Second, the network block we designed combines text features and predictive masks to learn the quality of text masks and regresses the intersection ratio of text masks to improve the quality of character masks. The model was tested on ICDAR2013, IC- DAR2015, and Total-Text datasets. Experimental results show that the text detection recall rate increases by 1.4% and 0.2% under the first two datasets, and the experimental results under three different vocabularies- is also show that MS TextSpotter has higher accuracy than other text recognition models and is more suitable for curved text recognition in natural scenes.

Keywords: Curved text · Text detection · Text recognition · Semantic segmentation · End-to-end training.

1 Introduction

Natural scene text detection and recognition provides a fast and automatic way to access text information existing in natural scenes, which is conducive to the re-

* This work was supported in part by the National Natural Science Foundation of China under Grant 61963017; in part by Shanghai Educational Science Research Project, China, under Grant C2022056; in part by Shanghai Science and Technology Program, China, under Grant 23010501000; in part by Humanities and Social Sciences of Ministry of Education Planning Fund, China, under Grant 22YJAZHA145.

Fig. 1. Classification score cls and comprehensive score mask of test results

alization of various applications in life.e, such as verification code recognition[1], label recognition[19]. However, natural scene text has great differences in font size, arrangement direction, font type, text sparseness, etc.

In recent studies, instance segmentation is often used in an end-to-end text recognition framework, but the direct use of instance segmentation networks to detect text has certain defects: in the character segmentation stage, the score of the text mask is shared with the confidence of the text classification. Mask TextSpotter [3] regards the text classification confidence as the text mask score, but the real text mask quality is quantified as the IoU(Intersection over Union) of the instance mask and its corresponding ground truth value, usually the true mask score does not correlate well with the classification score. As shown in Figure 1, there is a difference between the text classification confidence Scls and the text mask score Smask. The text instance classification prediction results in accurate box-level positioning results and high classification confidence, but their corresponding character mask scores have certain difference. Thus it is inappropriate to use the confidence of text classification to measure the quality of the mask.

To solve this problem, the MS TextSpotter (Mask Scoring TextSpotter) model is proposed in this paper, the model introduces the instance segmentation module and character mask Intersection over the Union module, which not only combines text features with prediction mask, learns the quality of text mask to regress the intersection ratio of text mask, improves the quality of character mask, but also realizes the detection of multi-direction and curved text in the natural scene. Experimental results show that, compared with the previous model, MS TextSpotter has a significant improvement in detection efficiency and recognition accuracy.

In summary, the main contributions of this paper are as follows:

1) Text detection and recognition can be realized by means of semantic segmentation, and curved text in natural scenes can be detected and recognized.

2) The character mask scoring mechanism is used to improve the integrity of character mask, which integrates semantic category and integrity of character mask. The model improves the quality of character mask.

3) It is proposed that the text detection and recognition model is based on end-to-end, which reduces the computational redundancy

The remainder of this paper is organized as follows:In Section 3, we describe the proposed method in detail, include character mask scoring mechanism and text recognition model. Experimental results are presented in Section 4, Finally, some conclusion remarks and future works are given in Section 5.

2 Related Work

As an important research direction of machine vision, text detection of natural scenes has been put into a lot of research in recent years. Before the advent of deep learning, the main trend of scene text detection is bottom-up, mainly using hand-made features.

The end-to-end natural scene text recognition method combines text detection task and text recognition task in a unified network model. This method usually shares the underlying convolution features, detects the text region according to the shared features, and then feeds the shared features of the text area to the recognition module to recognize the text content. For example, Mask TextSpotter [3] proposed a model that can detect and recognize arbitrary shape text instances. Although Mask TextSpotter [3] has achieved good results, but in the character segmentation stage, the score of text mask is shared with the confidence degree of text classification. The confidence degree of text classification is used to measure the quality of mask, which reduces the accuracy of text detection.

The MS TextSpotter model proposed in this paper,which introduces the Mask Head module into the network to realize the detection of multi-directional and curved text in the natural scene, and introduces the MaskIoU Head to learning the quality of the text mask, the cross merge ratio of the text mask is regressed by learning the quality of the text mask. The quality of the character mask is improved, meanwhile the accuracy of the text recognition is improved.

3 Our Proposed MS TextSpotter

The MS TextSpotter model is proposed to detect the text of the input image and convert all the detected text into the corresponding text sequence. The model text detection module uses the target detection model Fast R-CNN [9] to detect the horizontal rectangular area of the text. According to the characteristics of the text area, the text recognition module outputs the text instance probability map, character (English characters and numbers) instance probability map and character background probability map. Finally, the pixel voting algorithm is used to construct the character sequence from left to right.

3.1 Architecture

The overall architecture of the network is shown in Figure 2, which is improved based on Mask R-CNN [14], It can be seen that the network is composed of the following components: FPN(Feature Pyramid Network) [15] for image feature extraction, RPN(Regional Proposal Network) [10] for generating text region

suggestions, R-CNN for boundary box regression, a Mask Head branch for text segmentation and character segmentation, and a MaskIoU Head branch for character mask scoring. The following describes the role of each module.

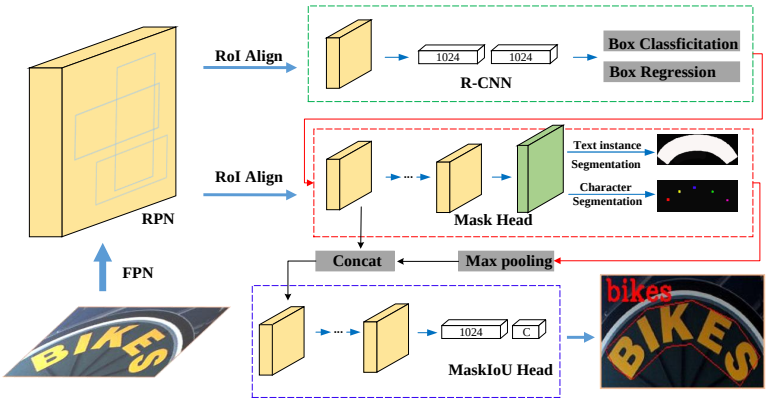

Fig. 2. MS TextSpotter model overview. The solid arrows mean the data flow both in training and inference period, The FPN network is shown in Figure 3

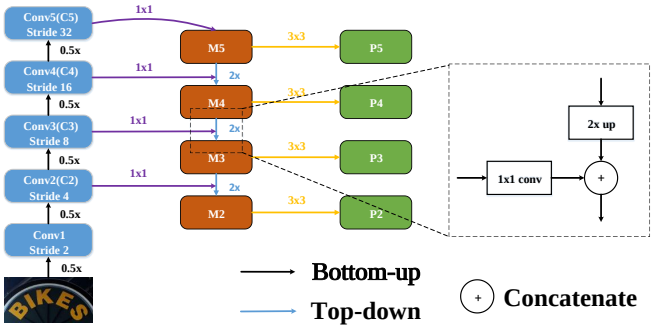

Fig. 3. Feature pyramid network

Backbone The text in the natural scene is complex and diverse, there are different sizes, and texts of different sizes correspond to different characteristics. The low-level feature semantic information is less, but the object location is accurate. High-level feature semantic information is richer, but the target location

is relatively rough.Therefore, text contour, texture and edge can be extracted by low-level features, and text details can be processed by high-level features. As shown in Figure 3, in order to construct high-level semantic feature maps at various scales, MS TextSpotter adopts a feature pyramid structure with ResNet-50 as the backbone. For single scale image input, FPN uses a top-down architecture to fuse the characteristics of different resolutions, which improves the accuracy at marginal cost.

RPN Ms TextSpotter generate text suggestion regions for R-CNN, Mask Head and MaskIoU Head branches through RPN. Referring to paper[14], we set different anchors at different stages according to the size of anchors.We use RoI Align [14] to uniformly represent the features of the Bounding box generated by RPN. Compared with RoI pooling, RoI Align retains more accurate position information, which is very important for the segmentation task in Mask branches.

R-CNN The input of R-CNN branch is generated by RoI Align according to RPN. This branch includes two tasks: Bounding box classification and Bounding box regression. Its main purpose is to provide more accurate location information for the detected text area.In the text detection based on R-CNN, the detection problem is treated as a classification problem and regressed into a more accurate text detection box.

Mask Head The Mask Head branch is mainly responsible for three tasks: text instance segmentation, character instance segmentation and background instance segmentation, as shown in Figure 4. After inputting a ROI feature of size, it is passed through three convolutional layers and one deconvolution layer in turn, and the 38-dimensional probability map is output by the last convolutional layerIt includes 36 character instance probability map, one global text instance probability map and one character background probability map. Among them, the text probability map is used to predict the text instance area in the rectangular area; the 36 character probability map includes 26 English letters and 10 numbers to predict different character regions in the rectangular area; the character background probability map is used to predict the non text area in the rectangular area.

MaskIoU Head MaskIoU Head combines the feature of character instances with their corresponding character masks, regresses the cross merge ratio of character masks, and corrects the deviation between the quality of character masks and the score of character masks by re-scoring strategy.

The score of character mask prediction as S_{mask} , an ideal S_{mask} value equal to the pixel-level IoU between the predicted character mask and the corresponding ground truth mask. Because each mask belongs to only one class, the ideal one can only have positive values for the category with ground truth value, and

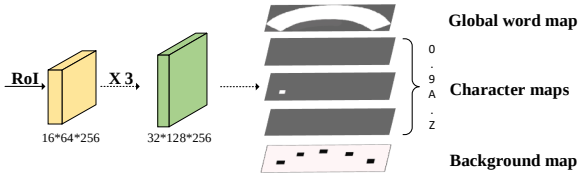

Fig. 4. Illustration of the Mask Head branch. The branch consists of four convolution layers and one de-convolution layer. The last layer produces 38 channels prediction map, including global text map, 0-9 and A-Z character map and background map

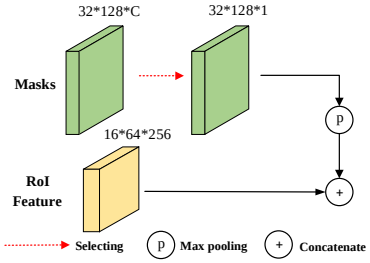

Fig. 5. The input of MaskIoU head,ROI Feature represents the features of the RoI Align layer, and masks represents the features of the prediction mask

the score for other categories is zero. MS TextSpotter divides learning tasks into mask classification and MaskIoU regression. All object categories are represented as follows: $S_{\text{mask}} = S_{\text{cls}} \times S_{\text{iou}}$, Among them, S_{cls} focuses on mask classification ,which has been completed in the task of Mask Head branch, so we can get the corresponding classification score directly. The main task of this stage will be the return of S_{iou}.

The input of MaskIoU Head is composed of the features of the RoI Align layer and the features of the prediction mask. As shown in Figure 5, in order to connect the RoI Feature with the masks, masks need to pass a max pooling layer with a convolution kernel size of 2 and a step size of 2 to obtain the same dimension as the ROI Feature. In the MaskIoU Head branch, there are 4 convolutional layers with a convolution kernel size of 3 and a channel number of 256, and 3 fully connected layers. The 3 fully connected layers follow the Fast R-CNN [9], the output of the first two fully connected layers is 1024, and the output of the last fully connected layer is the number of categories.

3.2 Label generation

For the image input in training, the general ground truth value includes $P = \{p_1, p_2, \cdots, p_m\}$ and $C = \{C_1, C_2, \cdots, C_n\}$, $C_n = (cc_n, cl_n)$. p_i represents the text instance area and is composed of a polygon box. cc_j and cl_j represent the

corresponding location and category of the character pixel. Firstly, cover the polygon with the smallest horizontal rectangle, and then generate objects for RPN network and Fast R-CNN network according to the paper [10], Here, we need to generate two types of targets for mask branch: global map for text instance segmentation and character map for character semantic segmentation using ground truth value P, C and suggested area provided by RPN. Given a suggested region r, we use the matching mechanism of paper[13] to obtain the best matching horizontal rectangle. The corresponding polygons and characters can be further obtained. Next, the matching polygon and character boxes are moved and resized to align the proposed area. $H \times W$ of the object map is calculated according to the following equation.

$$B_x = (B_{x0} - \min(r_x)) \times W/(\max(r_x) - \min(r_x)) \tag{1}$$

$$B_y = (B_{y0} - \min(r_y)) \times H/(\max(r_y) - \min(r_y)) \tag{2}$$

Where (B_x, B_y) and $((B_{x0}, B_{y0}))$ respectively represent the updated polygon vertexes and the original polygon vertexes. (r_x, r_y) is the vertexes of the proposal r. After that, just normalize the polygon on the mask initialized to zero and fill the polygon area with a value of 1. The generation of character map is as follows: first, we shrink all character bounding boxes by fixing the center point of the character bounding box and shortening its edge to a quarter of the original edge. Then, the pixel value in the shrunk character bounding boxes is set to its corresponding category index, and the pixel value outside the shrunk character bounding box is set to 0. If there are no character bounding boxes annotations, all values are set to -1.

3.3 Optimization

The MS TextSpotter model we propose is multi-task. Compared with the loss function designed by Mask RCNN, we add a global text instance segmentation loss and character segmentation loss. The loss function is as follows:

$$L = L_{\text{rpn}} + \alpha_1 L_{\text{cls}} + \alpha_2 L_{\text{box}} + \alpha_3 L_{\text{global}} + \alpha_4 L_{\text{char}} \tag{3}$$

Where, L_{rpn}, L_{cls} and L_{box} are the loss functions of RPN and Fast RCNN, L_{global} and L_{char} reference paper[3], representing the instance segmentation loss and character segmentation loss.

$$L_{\text{global}} = -\frac{1}{N}\sum_{n=1}^{N}[y_n \times \log(S(x_n)) + (1 - y_n) \times \log(1 - S(x_n))] \tag{4}$$

For L_{global}, N represents the total number of pixels in the global text map, $y_n (y_n \in (0, 1))$ represents the pixel label, and x_n represents the output pixels.

$$L_{\text{char}} = -\frac{1}{N}\sum_{n=1}^{N} W_n \sum_{t=0}^{T-1} Y_{n,t} \log(\frac{e^{X_{N,t}}}{\sum_{k=0}^{T-1} e^{X_{n,k}}}) \tag{5}$$

$$w_i = \begin{cases} 1 & \text{if } Y_{i,0} = 1 \\ N_{\text{neg}}/(N - N_{\text{neg}}) & \text{otherwise} \end{cases} \tag{6}$$

For L_{char}, T represents the number of categories, N represents the number of pixels in each map, in which the output map X can be seen as a $N \times T$ matrix. Y corresponding to the ground truth value X, the weight W is used to balance the loss value of character class and background class, and N_{neg} represents the number of background pixels, and its weight can be calculated by equation (6).

4 Performance analysis

Our proposed MS TextSpotter model can detect arbitrary shape text in natural scenes. We test its performance on three datasets, ICDAR2013, ICDAR2015, and Total-Text. Here is a brief description of all these related datasets.

Fig. 6. Visual reter for text detection and recognition on ICDAR2013 dataset.

Fig. 7. Visual results of MS TextSpotter for text detection and recognition on ICDAR2015 dataset

Fig. 8. Visual results of MS TextSpotter for text detection and recognition on Total-Text dataset

4.1 Horizontal text

For horizontal text, we evaluate our model on the ICDAR2013 dataset. Firstly, the short edge length of the input image is uniformly set to 1000 pixels. Secondly, our model is compared with six detectors, including FCRNall+multi-filt [23], Textboxes [22], Deep TextSpotter [24], Li et al. [25],Mask TextSpotter [20],Text Perceptron[21], the comparison of the detection results between our model and other scene text recognition models is shown in Table 1 and Table 3. Strong, Weak and Generic mean a small lexicon containing 100 words for each image, a lexicon containing all words in the whole test set and a large lexicon respectively. Specifically, it can be seen from Table 3 that even if it is only detected on a single scale, MS TextSpotter outperforms some of the previously proposed methods [20, 21, 25] under the three indicators of Precision, Recall and F-Measure, especially In terms of recall rate, compared with 89.5%of the most advanced detection method Mask TextSpotter [20], it exceeds 1.4 points, and the recall rate reaches 90.9%. As shown in Table 1, in the ICDAR2013 data set test, based on the End-to-End evaluation method, MS TextSpotter outperforms other advanced models in three different constraint vocabularies. the evaluation method based on Word Spotting, It is only slightly lower than the most advanced Text Perceptron in Strong's constraint vocabulary, and it is better than the best method proposed before under Weak and Generic.

4.2 Oriented text

For multi-directional text in natural scenes, MS TextSpotter evaluates its performance on the ICDAR2015 dataset. First, the short side length of the input image is uniformly set to 1600 pixels, and then MS TextSpotter is compared with 7 detectors, including TextSpotter [28], StradVision , Deep TextSpotter [26], Mask TextSpotter [3], Char- Net [4], Text Perceptron [5], TextDragon [6]. The recognition results of Ms TextSpotter and the comparison results with other text recognition models are shown in Tables 2 and 3. As shown in Table 3, in terms of recall rate, MS TextSpotter once again showed excellent results. Compared with detectors [3, 5, 12], it has been significantly improved, and it is better

Table 1. MS TextSpotter's text recognition results on the ICDAR2013 dataset and comparison with other text recognition models, the best results are marked in bold.(S for Strong, W for Weak and G for Generic.)

Method	Word Spotting			End-to-End			Speed
	S	W	G	S	W	G	FPS
FCRNall+multi-filt [23]	–	–	84.7	–	–	–	–
Textboxes [22]	93.9	92.0	85.9	91.6	89.7	83.9	1.0
Deep TextSpotter [24]	92	89	81	89	86	77	**9**
Li et al. [27]	94.2	92.4	88.2	91.1	89.8	84.6	1.1
Mask TextSpotter [20]	92.7	91.7	87.7	93.3	91.3	88.2	3.1
Text Perceptron [21]	**94.9**	94.0	88.5	91.4	90.7	85.8	–
MS TextSpotter	94.6	**94.1**	**88.7**	**94.8**	**92.1**	**88.7**	2.9

Table 2. MS TextSpotter's text recognition results on the ICDAR2015 dataset and comparison with other text recognition models, the best results are marked in bold.(S for Strong, W for Weak and G for Generic.)

Method	Word Spotting			End-to-End			Speed
	S	W	G	S	W	G	FPS
TextSpotter [28]	37.0	21.0	16.0	35.0	20.0	16.0	1.0
Stradvision	45.9	–	–	43.7	–	–	–
Deep TextSpotter [26]	58.0	53.0	51.0	54.0	51.0	47.0	**9.0**
Mask TextSpotter [3]	82.4	78.1	73.6	83.0	77.7	73.5	2.0
Char Net [4]	–	–	–	83.1	**79.1**	69.1	–
Text Perceptron [5]	84.1	79.4	67.9	80.5	76.6	65.1	–
TextDragon [6]	**86.2**	**81.6**	68.0	82.5	78.3	65.1	–
MS TextSpotter	85.4	80.0	**75.6**	**84.6**	78.9	**74.6**	1.9

than the most advanced method Char-Net, the recall rate reached 90.6%. As shown in Table 2, in the ICDAR2015 dataset test, based on the End-to-End evaluation method, MS TextSpotter achieved the best results in the two different constraint vocabularies of Strong and Generic. The performance in the table is only lower than Char-Net.

Table 3. Results of MS TextSpotter text detection on datasets ICDAR2013 and ICDAR2015

Method	**ICDAR2013**				**ICDAR2015**			
	P	**R**	**F-M**	**FPS**	**P**	**R**	**F-M**	**FPS**
CTPN [11]	93.0	83.0	88.0	7.1	74.0	52.0	61.0	–
Seglink [29]	87.7	83.0	85.3	**20.6**	73.1	76.8	75.0	–
EAST [13]	–	–	–	–	83.3	78.3	80.7	–
SSTD [30]	89.0	86.0	88.0	7.7	80.0	73.0	77.0	**7.7**
Wordsup	93.3	87.5	90.3	2.0	79.3	77.0	78.2	2.0
He et al. [12]	92.0	81.0	86.0	1.1	82.0	80.0	81.0	1.1
Mask TextSpotter[3]	94.8	89.5	92.1	3.0	86.6	87.3	87.0	3.1
Char Net[4]	–	–	–	–	**92.6**	90.4	**91.5**	–
Text Perceptron[5]	94.7	88.9	91.7	10.3	92.3	82.5	87.1	**8.8**
TextDragon[6]	–	–	–	–	92.4	83.7	87.8	–
MS TextSpotter	**95.1**	**90.9**	**92.9**	2.9	89.0	**90.6**	89.8	2.8

4.3 Curved text

For curved text, MS TextSpotter evaluates its performance on the Total-Text dataset. First, the short side length of the input image is uniformly set to 1000 pixels, and then MS TextSpotter is compared with 6 detectors, including Ch'Ng et al. [18], Liao et al. [7], Mask TextSpotter [3], Char-Net [4], TextDragon [6], the comparison of the detection results of MS TextSpotter and other detectors is shown in Table 4.

Compared with the most advanced model, the results show that MS TextSpotter performs better in the detection and recognition of curved text, and the accuracy, recall, and average harmony have been significantly improved. It can be seen from Table 4 that although MS TextSpotter's detection performance is inferior to the most advanced Char-Net (4.7%-6.4%), it is better than other text recognition models, and under the end-to-end evaluation method, MS TextSpotter has improved significantly, intersecting with Char-Net by 8.9%. It can be seen that the improvement of text detection accuracy mainly comes from more accurate positioning output, that is, using polygons instead of horizontal rectangles to detect text areas. The improvement of text detection recall rate mainly

Table 4. Results of MS TextSpotter for text detection and end-to-end recognition on dataset Total-Text

Method	Detection			End-to-End
	P	**R**	**F-M**	**F-M**
Ch'ng et al. [18]	40.0	43.0	36.0	–
Liao et al. [8]	62.1	45.5	52.5	48.9
Mask TextSpotter [3]	69.0	55.0	61.3	71.8
Char Net [4]	88.0	85.0	86.5	69.2
TextDragon [6]	85.6	75.7	80.3	74.8
MS TextSpotter	**72.3**	**64.2**	**68.0**	**75.8**

comes from the scoring of character masks, and correct scoring brings correct text detection and character segmentation.

4.4 Speed

Most existing natural scene text detection and recognition models detect and recognize text in a multi-step manner, which makes them difficult to run efficiently. Compared with these models, Mask TextSpotter has a good compromise between speed and accuracy. On the ICDAR2013 and ICDAR2015 datasets, the model can detect text at 2.9 FPS and 2.8 FPS, respectively. The speed is weaker than Text Perceptron [5], but MS TextSpotter achieves the highest detection accuracy.

4.5 Engineering Applications

With the development of economy, China's demand for railway freight is growing. In order to better match the train number with the train information management system, so as to carry out cargo interaction more quickly and accurately inform the train conductor of the running information in the process of running. The identification of train number is also becoming more and more important. The identification of train number is becoming more and more important.

At present, most of the processing steps of this kind of problem can be roughly divided into the following methods, including image pretreatment, location, segmentation and extraction of vehicle number, but the input image requires good lighting conditions, and the vehicle number is relatively easy to locate. In practice, the number pattern of freight train is quite different from the traditional car number pattern.

Therefore, we can integrate the photos of train numbers collected into our MS TextSpotter. Firstly, the train number was preprocessed, including correcting the tilted photos and filtering out the seriously distorted ones, and then the filtered pictures were input into our MS TextSpotter. In the network, RPN and

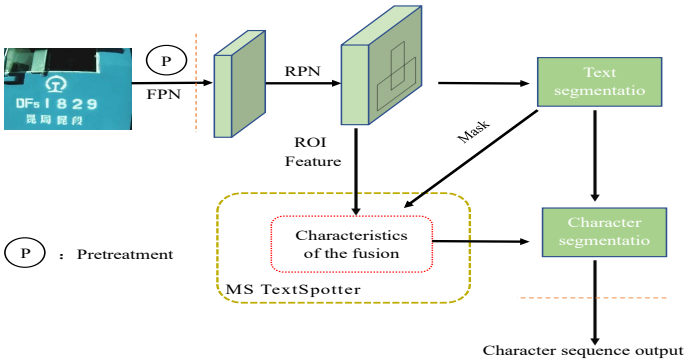

Fig. 9. Overall drawing process

RoIAlign are used to generate a feature map of a specific size. In the Mask Head stage, the input character map is converted into character sequence through pixel processing algorithm. In the MaskIou Head stage, the text Mask and RoI Feature obtained in the Mask Head stage are taken as input, and the final text recognition is obtained through a series of convolutional layers and full-connection layers, as shown in Figure 9 TextSpotter can effectively improve the accuracy of text detection and greatly reduce computational redundancy.

5 Conclusion and the future work

In this paper, we propose MS TextSpotter, an end-to-end network for text detection and recognition in natural scenes. It can effectively detect text and segment characters in complex background. Compared with the previous model, our proposed network is very easy to train and can detect and recognize curved text. In all experiments, MS TextSpotter has achieved excellent performance in horizontal text, multi-directional text, curved text and other datasets, which improves the recognition accuracy and greatly reduces false positives. Our model shows high efficiency and robustness in text detection and end-to-end recognition. In the future work, we will try to optimize our model to improve the speed of text detection in order to realize the application in real life. Secondly, we will explore the recognition of Chinese text.

Author Contributions Conceptualization, Y.L. and Y.Z.; Investigation, Y.Z., Y.L., H.Z.; Software, Q.Q.; Writing-Original Draft Preparation, Y.Z. and Y.L.; writing–review and editing, H.Z.; Funding Acquisition, Y.L. J.W.and H.Z.

Conflicts of Interest The authors declare no conflict of interest.

References

1. Gupta N, Jalal A S. Traditional to transfer learning progression on scene text detection and recognition: a survey[J]. Artificial Intelligence Review, 2022: 1-46.
2. Tian X.; Wang Z.; Wang J., Text detection of food labels based on semantic segmentation[J]. *Transactions of the Chinese Society for Agricultural Machinery*, **2020**, *51(8)*: 336-343.
3. Lyu P, Liao M, Yao C, et al. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018:67-83.
4. Liu W, Chen C, Wong K K. Char-Net: A Character-Aware Neural Network for Distorted Scene Text Recognition.[C]//Association for the Advancement of Artificial Intelligence. 2018, 1(2):4-12.
5. Qiao L, Tang S, Cheng Z, et al. Text Perceptron: Towards End-to-End Arbitrary-Shaped Text Spotting[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020,34(7):11899-11907.
6. Feng W , He W , Yin F , et al. TextDragon: An End-to-End Framework for Arbitrary Shaped Text Spotting[C]// 2019 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 2020.
7. Zheng, Y.; Li, Q.; Liu, J.; Liu, H.; Li, G.; Zhang, S., A cascaded method for text detection in natural scene images. *Neurocomputing*. **2017**, *238*, 307-315.
8. Liu W , Wang X , Bai X , et al. TextBoxes: a fast text detector with a single deep neural network. AAAI Press, 2017.
9. R. Girshick. Fast R-CNN[J]. IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440-1448.
10. Ren, S.; He, K.; Girshick, R.; Sun, J., Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems, 2015; pp. 91-99.
11. Tian Zhi, Weilin Huang, Tong He, et al. Detecting Text in Natural Image with Connectionist Text Proposal Network.[J]. CoRR, 2016, abs/1609.03605.
12. S. Q, R. M. Cascaded Segmentation-Detection Networks for Word-Level Text Spotting: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)[C], 2017.
13. Zhou X , Yao C , Wen H , et al. EAST: An Efficient and Accurate Scene Text Detector[J]. IEEE, 2017.
14. He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017:2961-2969.
15. Lin T Y, Dollár P, Girshick R, et al. Feature Pyramid Networks for Object Detection[C]//The IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017: 2117-2125.
16. Gupta A, Vedaldi A, Zisserman A. Synthetic Data for Text Localisation in Natural Images[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016:2315-2324.
17. Karatzas D, Shafait F, Uchida S, et al. ICDAR 2013 robust reading competition[C]//Document Analysis and Recognition (ICDAR), 2013 12th International Conference on. 2013.
18. Karatzas D, Gomez-Bigorda L, Nicolaou A, et al. ICDAR 2015 competition on robust reading[C]//2015 13th International Conference on Document Analysis and Recognition (ICDAR). IEEE, 2015:1156-1160.

19. Ch'Ng C K, Chan C S. Total-text: A comprehensive dataset for scene text detection and recognition[C]//2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). IEEE, 2017:935-942.
20. Lyu P, Liao M, Yao C, et al. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018:67-83.
21. Qiao L, Tang S, Cheng Z, et al. Text perceptron: Towards end-to-end arbitrary-shaped text spotting[J]. arXiv preprint arXiv:2002.06820, 2020.
22. Liao M, Shi B, Bai X, et al. Textboxes: A fast text detector with a single deep neural network[C]//Thirty-First AAAI Conference on Artificial Intelligence.San Francisco, California, USA, 2017: 4161-4167.
23. Gupta A, Vedaldi A, Zisserman A. Synthetic data for text localisation in natural images[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2315-2324.
24. Busta M, Neumann L, Matas J. Deep textspotter: An end-to-end trainable scene text localization and recognition framework[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017:2204-2212.
25. Li H, Wang P, Shen C. Towards end-to-end text spotting with convolutional recurrent neural networks[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017:5238-5246.
26. Busta M, Neumann L, Matas J. Deep textspotter: An end-to-end trainable scene text localization and recognition framework[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017:2204-2212.
27. Li H, Wang P, Shen C. Towards end-to-end text spotting with convolutional recurrent neural networks[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017:5238-5246.
28. Neumann L, Matas J. Real-time lexicon-free scene text localization and recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 38(9):1872-1885.
29. Shi B, Bai X, Belongie S. Detecting Oriented Text in Natural Images by Linking Segments[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017:2550-2558.
30. He P, Huang W, He T, et al. Single Shot Text Detector with Regional Attention[C]// International Conference on Computer Vision, 2017:3066-3074.
https://vision.in.tum.de/data/datasets/rgbd-dataset.