Researches advanced in Natural Scenes Text Detection Based on Deep Learning

Qingyang Zhao*

Shandong University, Weihai, Shandong province, 264209, China

*Corresponding author: 201900820153@mail.sdu.edu.cn

Abstract. The research on text detection and recognition in natural scenes is of great significance for obtaining information from scenes. Thanks to the rapid development of convolutional neural networks and the continuous proposal of scene text detection methods based on deep learning, breakthroughs have been made in the recognition accuracy and speed of scene texts. This paper mainly sorts, analyzes and summarizes the scene text detection method based on deep learning and its development. Firstly, the related research background and significance of scene text detection are discussed. Then, the second part is the elaboration of some main technical research routes of scene text detection. According to the timeline of the detection methods, the specific contents of various text detection models are further introduced. Thirdly, this paper compares and analyzes the experimental results of different models. Furthermore, improvements of some models with relationship, effects, advantages and disadvantages and expectations are further introduced. Finally, the challenges and development trends of scene text detection technology based on deep learning are summarized.

Keywords: text detection; Natural Scenes; Deep learning.

1. Introduction

Natural scenes text recognition refers to detecting and recognizing image text information in a special scene with unconstrained environment. Current natural scene text recognition consists of two steps, respectively are text detection and text recognition. Text detection is about using image processing technology and visual processing technology to extract and locate information of the text instances in images and text recognition is that using dictionary to match, judge to get the text content. The text detection and recognition technology in the natural scene images is beneficial to the development of human-computer interaction. And they can capture, analyze and understand content information hidden in the scene image. All in all, natural scenes text recognition is significance for improving image retrieval ability, industrial automation level, scene understanding ability and has been applied to the fields of detection and recognition, big data, image retrieval and intelligent robots and other intelligent products. It is very closely related that two steps of text detection and text recognition. As a pre-task of text recognition, text detection influences even decides the results of text recognition directly. Therefore, text detection and recognition of natural scene has already become a hotspot in the fields of computer vision and pattern recognition, document analysis and recognition.

As a part of text recognition, text detection is also a research hotspot. Early text detection mainly relied on three points to detect image text location: hand-designed complex features, classifiers, post-processing pipelines. A representative work is to obtain text candidates by detecting some of the largest stable extrema regions in an image. Alternatively, some methods use a sliding window to detect character regions, then obtain character content by using character confidence, finally adopt spatial constraints between characters to obtain text word content. Different from the regularity of texts which are in document images, texts which are in natural scene usually have many differences on the characteristics of font size, font category, arrangement direction, font color, and text sparsity. At the same time, affected by factors such as different light intensities, complex backgrounds and camera angles, the research on text detection and recognition technology in natural scenes has great resistance. At present, traditional text detection techniques cannot be applied to text recognition in

natural scene images with complex environmental backgrounds, a large number of oblique curved or even irregular text lines, etc.

With the rapid development of deep learning and artificial neural networks, natural image text detection and recognition based on convolutional neural network has already become a hot research in the field of current literature processing and analysis. Because of the advantage of deep learning which is that the model based on it is hierarchical and has many parameters, scene text detection based on deep learning can generate more abstract high-level representation and features by combining low-level features, and it has better results and detection accuracy. Besides, neural network which mainly consists of the convolutional neural networks (CNN) and the recurrent neural networks (RNN) also is helpful for the optimization and training of scene text detection network architecture. As a part of neural network, CNN uses the convolution operation to extract data features like the grid structure and the pooling layer retains the main features to prevent overfitting, respectively. RNN can be divided into three parts: input layer, hidden layer, output layer. And the nth input of the hidden layer consists of the nth input of the hidden layer and the nth-1st output of the hidden layer. By sharing parameters at different time points, the network has a backward connection in time and can learn to have Time series data characteristics and rules. More and more researchers use CNN-based model to extract image features and use RNN-based technology to detect the scene text, which promotes the continuous progress and development of this field.

Focusing on the mainstream frameworks in the field of text detection, in this paper, the related research background and significance of scene text detection are discussed. Then, the second part is the elaboration of some main technical research routes of scene text detection. According to the timeline of the detection methods, the specific contents of various text detection models are further introduced. Thirdly, this paper compares and analyzes the experimental results of different models. Furthermore, improvements of some models with relationship, effects, advantages and disadvantages and expectations are further introduced. Finally, the challenges and development trends of scene text detection technology based on deep learning are summarized.

2. Deep learning-based text detection method in natural scenes

Natural scene text detection is a necessary step for recognition. In the past 10 years, because of the rapid development of text detection with computer vision, a series of text detection algorithms based on deep learning have emerged. Text detection is a specific field of target detection research content. Many researchers draw on the technical ideas of target detection and combine deep learning technology with text detection. According to the difference of basic frames, some representative natural scene text detection algorithm mainly consists of the regression-based and segmentation-based, which can be seen in Table 1.

Table 1 Classification of representative text detection methods

model	method	time	text type	core
CTPN	Regression	2016	horizontal	CNN+RNN
EAST	Regression	2017	horizontal +rotate	FCN+LANMS
SegLink	Regression	2017	horizontal+curved	CNN+SSD (default box)
TextBoxes	Regression	2017	horizontal	CNN+SSD (default box) +NMS
TextBoxes++	Regression	2018	horizontal+rotate	CNN+SSD (default box) +NMS
PSENet	Segmentation	2019	horizontal+curved+irregular	PSE(BFS)
MSR	Regression+ Segmentation	2019	horizontal+curved+irregular	MSN
PAN	Segmentation	2019	horizontal+curved+irregular	FPEM+FFM +PA
ABCNet	Regression	2020	horizontal+rotate+irregular	Bezier Curve + BezierAlign
FCENet	Regression	2021	Any	FPN+IFT
SwinTextSpotter		2022	Any	FPN+ Swin-Transformer

2.1. CTPN

Connectionist Text Proposal Network (CTPN) is one of the most representative text detection methods, whose framework can be seen in Figure 1 [1]. CTPN gets feature maps from the conv5 of VGG16 and uses 3×3 window to slide the feature maps (predict the categories and locations corresponding to the $k$anchors at the windows' position). Then all the features of windows in each row will be input into the BLSTM. Every two similar proposals match, and merge different pairs until no common elements. CTPN has some contributions, which are described below. Vertical anchor mechanism (jointly predicts location and text/non-text score of each fixed-width proposal, considerably improving localization accuracy). Top-down methods (detect the text area first, then find the text line). Seamless combination of CNN and RNN (CNN is used to extract deep features, and RNN is used for sequence feature recognition). The two are seamlessly combined and have excellent performance in detection.

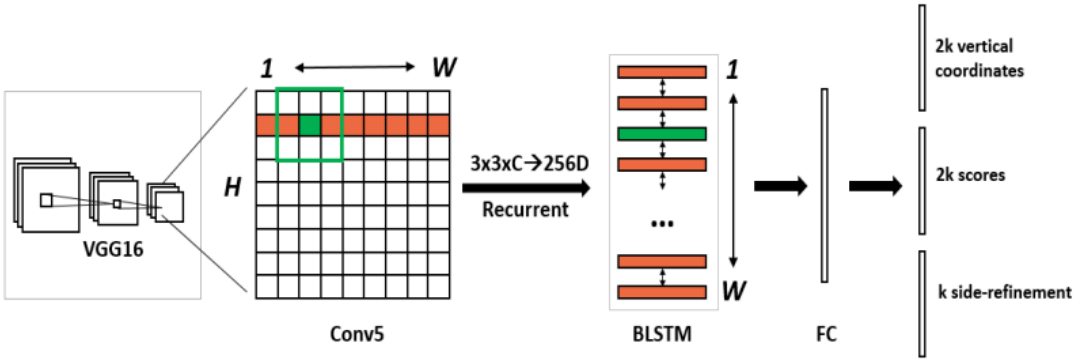

Figure 1 Structure of CTPN network [1]

2.2. EAST

In 2017, the Efficient and Accurate Scene Text Detector (EAST) is proposed, which only consists of FCN and NMS [2]. FCN directly form the text area without other redundant and time-consuming step. Use locality-aware NMS to filter the results. Flexibility to build text areas forecasts, and their

geometric shape can be RBOX (rotated box) or QUAD (quadrilateral). Specifically, as the Figure 2 shown, the key module of EAST is the QUAD and RBOX training sample generation. QUAD is defined as the point p1 are $(x1, y1)$, the point p2 are $(x2, y2)$, the point p3 are $(x3, y3)$, and the point p4 are $(x4, y4)$. And actually, QUAD to RBOX is methods from $[x1, y1, x2, y2, x3, y3, x4, y4]$ to [theta, vx1, vy1, vx3, vy3]. Thirdly, Loss Function. Calculate score map loss and geometry loss and calculate the total loss by weighting the two. Fourthly, Locality-Aware NMS. When the IoU of the two boxes is greater than the threshold, the similar text boxes are merged, and all the text boxes are traversed in turn for weighted merging. After the merging, the NMS non-maximum suppression judgment is performed, and the merged feasible set is obtained and stored. EAST have an effective and accurate result. However, because the receptive field is not large enough, EAST is not very effective for detecting long text lines and is more suitable for short text lines. Compared with CTPN, due to the existence of LSTM, CTPN has better detection effect on long text than EAST, but the detection effect on oblique text lines is not very good.

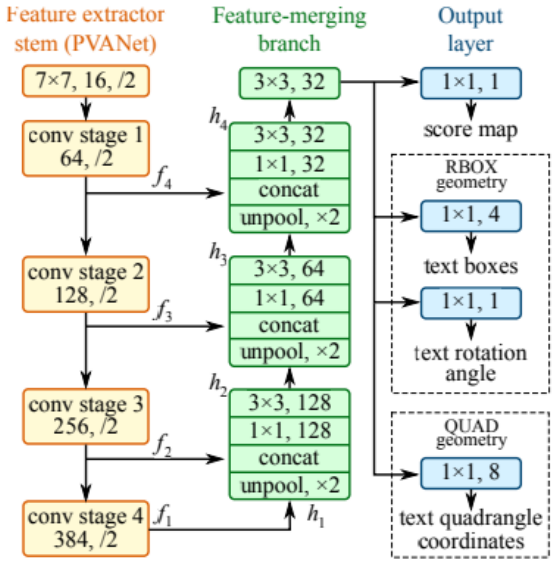

Figure 2 Structure of EAST network [2]

2.3. SegLink

SegLink is like the idea of CTPN, which first finds a part of the text line and then connects all the parts to form a complete text line [3]. As the Figure 3 shown, SegLink mainly includes the feature extraction module and segments links detection module. For the input, since the network all adopts the convolution structure, there is no requirement for the size of the input image. The output articles are called segments and links. Segments can be understood as small boxes one by one. These small boxes are like the default boxes in SSD. They may not necessarily be able to frame a word in a box but may only frame a part of a word. Links are to connect segments. To put it bluntly, it is a probability value of whether two boxes are the same text. The representation method of segment is b = (x_b, y_b, w_b, h_b, θ_b). The network needs to output the confidence of the segment and a regression offset of the segment relative to the five parameters of the default boxes. After calculating the information of default boxes, we need use the information output after the feature map is convolved to calculate the location of segment in image. Link detection includes Within-Layer Link detection and Cross-Layer Link detection. The first is Within-Layer Link detection which is a link connecting two adjacent segments and indicating the segments are classified in the same word or in the same frame. The role of link is not only to connect adjacent segments, but also to distinguish adjacent segments that do not belong to the same peer or the same calibration frame. Cross-Layer Link detection: in this paper, to detect text of different scales, feature maps of different layers are used to output segments, which will cause the output segments of different layers to be in the same position, but may only have different

sizes, which will eventually lead to problems in the merging of boxes generated by different layers. This problem you need to use Cross-Layer Link to solve it. Thirdly, Combining Segments with Links. First set a threshold to filter noise. After filtering, adopt DFS and connect them. Finally use algorithm to combine the connected results into text boxes. Compared with previous methods, SegLink improves the accuracy, speed, and ease of training. Besides, for multiple languages, it works well. SegLink performs well on level, selection, multilingual datasets, and it is accurate and flexible. However, it can't detect curved text and some text with widely space.

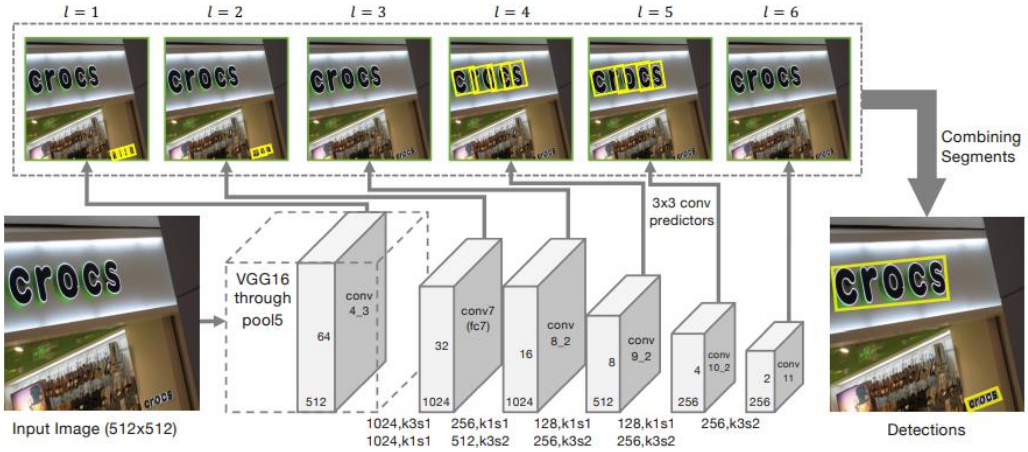

Figure 3 Structure of SegLink network [3]

2.4. TextBoxes and TextBoxes++

The network structure of TextBoxes is not complex. Firstly, remove the full connection in SSD and replace it with convolution [4]. Then, use 1*5 convolution kernel instead of 3*3 convolution kernel. In addition, using 1, 2, 3, 5, 7, and 10 in the proportion of default boxes. After the obtained default boxes, use the regression method like the regression method of SSD to regress and get the possible results. TextBoxes is fast, easy and gets an effective result with concision. However, the slender default boxes may be dense in the horizontal direction and sparse in the vertical direction so that the detection is inaccurate. TextBoxes++ continues some of TextBoxes and improves its network [5]. First, add a vertical offset to each default box to make detection more accurate. Second, TextBoxes++ use different ground truth to output. Third, TextBoxes++ also has some data enhancement methods. The improvement of TextBoxes++ makes detection for arbitrary-oriented scene text get a more accurate and faster result. Both use the similar methods of SSD to detect scene text and are faster and more concise than some other good methods.

2.5. PSENet

PSENet can precisely detect text in the scene by a pixel segmentation-based method [6]. PSENet use kernels and progressive scale expansion to get accurate and precise text areas in the scene even when some text lines are very close. The progressive expansion algorithm enables two text instances to be separated even when they are very close, and precisely locates the position of the segmented text instance. Firstly, input image and obtain four 256 channels feature maps from backbone network. Secondly, the four feature maps through the function can get the feature map F. Thirdly, expand the kernels of all instances into their full shapes in using progressive scale expansion algorithm and get R. PSENet is well done in the scene text detection. However, some hyperparameters like minimum scale ratio and number of segmentation results need change and select some suitable values when face different datasets.

2.6. Other representative methods

PAN proposes a computationally inexpensive segmentation head composed of FPEM and FFM, and innovatively uses PA to accurately aggregate text pixels as a learnable post-processing

implementation [8]. Lightweight segmentation and this pixel aggregation method make scene text detection more accurate and faster, even for difficult curved text, while maintaining good speed.

The detection framework of MSR is segmentation and nearest boundary point regression like that of EAST [7]. Different to EAST, MSR returns the XY distance between the pixels in the text center area and the nearest border point. Multi-scale Network which is proposed by MSR can not only predict the details of different levels, but also extract text features of different scales in the image. This method is contribution and creation, and it can provide a new view for the field of text detection. Compared with EAST, the regression method of MSR can effectively avoid the disadvantage, which the receptive field of the convolutional layer is limited in long texts. All in all, MSR has excellent tolerance to changes in text length and scale and has excellent detection results for different lengths, shapes and curvatures.

For text with highly curved shapes in the scene, FCENet introduces Fourier contour embedding to improve detection accuracy [10]. FCENet mainly consists of backbone, Feature Pyramid Network (FPN) and post-processing including Inverse Fourier Transform (IFT) and Non-Maximum Suppression (NMS). The FCENet method is unnecessary to introduce a complex post-processing so that it is easier to implement. FCENet can be divided into two stages, Resampling and Fourier Transformation. They are based on the ground truth point to obtain the dense point sequence and use the resampling point sequence to calculate the Fourier factor c_k. Combining circular motions of different fixed frequencies with c_k, the contour can be reconstructed. To the target texts in the image, FCENet firstly predicts compact Fourier signatures of texts, then get more precise text contours via IFT and NMS. The FCENet network structure adopts the typical Backbone + FPN as the backbone network. The feature map output by FPN will go through the shared prediction head for classification prediction and regression prediction. In the classification part, the network obtains the score map belonging to the text classification by predicting and multiplying the probability map of the text center and around regions. In the regressions part, it directly predicts the Fourier feature vector at each pixel position. In the post-processing process, the algorithm performs IFT on the Fourier feature vector to reconstruct the text contour on the area higher than the score threshold. And then uses NMS to filter out the text with high coincidence to ensure the detection effect.

3. Method evaluation and result analysis

3.1. Common scene text detection datasets

Common scene text detection datasets mainly include the ICDAR, the CTW1500, the MSRA-TD500, the Total-Text and the COCO-Text.

The ICDAR 2013 dataset consists of 229 training images and 233 testing images, with word-level annotations provided. It is the standard benchmark dataset for evaluating near-horizontal text detection. ↵

The ICDAR 2015 dataset has 1000 training images and 500 test images which are collected by Google Glass and suffers from low resolution and motion blur. All text instances are annotated at word level using quadrilateral boxes.↵

The CTW1500 consists of 1000 training images and 500 test images. There are 10751 multi-oriented text instances and 3530 of them are arbitrarily curved. Each text instance is annotated at text-line level, where texts are largely in English and Chinese.↵

The MSRA-TD500 consists of 300 training images and 200 test images. All captured text instances are printed in English and Chinese which are annotated at text-line level by using best-alighted rectangles. ↵

The Total-Text has 1255 training images and 300 test images. The texts of it are all in English and have many multi-oriented curved text instances annotated at word level. ↵

The COCO-Text has 63,686 images, 173,589 instances of text, both handwritten and printed, legible and non-legible. The file size is 12.58GB. Training set: 43686 pictures, test set: 10000 pictures, validation set: 10000 pictures.↵

Figure 4 common scene text detection datasets

3.2. Evaluation indicators

The performance indicators of scene text detection mainly include recall, precision, etc. Recall refers to the proportion of the number of correctly identified positive samples among all positive samples in the test set. Precision refers to the proportion of the number of correctly identified positive samples among the identified positive samples. In general, there is a negative correlation between precision and recall. Sometimes we need to comprehensively weigh these two indicators, which leads to a new indicator F-score. F-score is F1-score here and F1-score is equal twice recall times precision divided by the sum of recall and precision. In practical applications, according to different data sets and applications, the indicators for evaluating models are also different. Sometimes the larger the recall value is, the better, and sometimes the larger the precision value is, the better.

3.3. Performance comparison

Most text detection methods use ICDAR2013, ICDAR2015 and Total-Text as datasets. This section compares the experimental results of the main text detection methods on these datasets, as shown in Table 2. The same algorithm and the same network architecture have little difference in indicators such as recall and precision, but indicators such as running speed and inference time will be affected by some external factors. To ensure a certain degree of rigor, indicators such as FPS and inference time are not listed here and may only be compared when comparing some models.

Among the methods based on text component candidates, the F-measure of CTPN on the ICDAR2013 dataset reaches 0.88. However, due to its technical characteristics, CTPN has poor detection performance for oblique text, and the F-measure on the ICDAR2015 dataset is only 0.61. SegLink enhances the detection robustness of texts of different scales in the network model, overcomes the shortcoming that CTPN can only detect horizontal texts, and improves the F-measure of ICDAR2015 on the dataset by 0.14 compared with CTPN. PSENet adopts the method of multi-level text region detection to overcome the disadvantage of indistinguishable text instances in the case of poor image quality. The F-measure on the dataset ICDAR2015 reaches 0.87, which is better than other proposals based on candidate regions and based on semantic segmentation method.

Table 2 Performance of representative text detection methods

model	Time	datasets	backbone	result (rough)		
				recall	precision	F-score
CTPN	2016	ICDAR 2013	VGG16	/	/	0.88
		ICDAR 2015				0.61
EAST	2017	ICDAR 2015	VGG16 PVANET	0.72	0.80	0.76
		COCO-Text		0.31	0.45	0.37
		MSRA-TD500		0.64	0.82	0.72
SegLink	2017	ICDAR 2013	SegLink	0.83	0.87	0.85
		ICDAR 2015		0.73	0.77	0.75
		MSRA-TD500		0.70	0.86	0.77
TextBoxes	2017	ICDAR 2013		0.83	0.88	0.85
TextBoxes++	2018	ICDAR 2013	TextBoxes++_MS Quad_MS	0.84	0.91	0.88
		ICDAR 2015		0.78	0.88	0.83
		COCO-Text		0.56	0.61	0.58
PSENet	2019	ICDAR 2015	PSENet-1s ResNet	0.85	0.88	0.87
		ICDAR 2017 MLT		0.68	0.77	0.72
		SCUT-CTW1500		0.79	0.82	0.81
MSR	2019	ICDAR 2015	ResNet	0.78	0.86	0.82
		MSRA-TD500		0.76	0.87	0.81
		SCUT-CTW1500		0.78	0.85	0.81
		Total-Text		0.74	0.83	0.79
PAN	2019	ICDAR 2015	PAN PAN-640	0.80	0.84	0.82
		MSRA-TD500		0.83	0.85	0.84
		SCUT-CTW1500		0.81	0.86	0.83
		Total-Text		0.81	0.89	0.85
ABCNet [9]	2020	Total-Text	ResNet-50-FPN	0.81	0.88	0.84
		CTW1500		0.78	0.84	0.81
FCENet	2021	Total-Text		0.82	0.89	0.85
		CTW1500		0.83	0.87	0.85
SwinTextSpotter [11]	2022	ICDAR-2015	ResNet-50 Swin	/	/	0.84
		ReCTS		0.87	0.94	0.90
		RoIC13		0.72	0.84	0.77
		VinText		/	/	0.71
		Total-Text		/	/	0.88
		SCUT-CTW1500		/	/	0.88

4. Discussion

4.1. Existing problems

For the previous scene text detection algorithms, recall and precision have good performance, but the large amount of calculation and time-consuming are some problems that need to be solved. On the other hand, some recently proposed scene text detection methods are all lightweight algorithms, which can ensure accurate and better detection results while reducing the amount of calculation and shortening the calculation and reasoning time, which is exactly one direction of the current algorithm optimization.

Due to advances in technology and the popularity of scene text detection applications, many detection algorithms need to keep up with them. The popularity of smartphones and the application of many human-computer interactions have led to the need for faster results in scene text detection and recognition of the detected text, and then transmitted to the next link that is matched with practical applications. For example, in our daily work and life, photo translation as an app in a mobile phone may be needed by some people. They may need to take photos and translate apps to study, work, and communicate to reduce the burden of translating text. And this kind of app will undoubtedly need fast and accurate text detection and positioning, identify the positioned text to get the text content, and then translate it through some databases and semantically supervised sorting based on deep learning. Generally speaking, it takes about three seconds to complete the translation and display of the text, and the time required for scene text detection is particularly short. For the convenience of the public, the time-consuming of photo translation will also be reduced. If the target is one second, the time allocated for scene text detection may be within 0.3s, and it needs to be able to target multilingual texts. Therefore, the current problem of scene text detection is to broaden the language types and optimize the speed.

4.2. Future direction

For the future research direction of scene text detection, I think it is mainly divided into two parts: efficiency and language. The first is to maximize efficiency and reduce time-consuming while ensuring accuracy. The main purpose of this part of the research is to reduce the amount of computation, optimize the network structure, and eliminate redundant steps to make the network structure more streamlined and the connections between modules more elegant. The second is to broaden the types of text detection languages. Some languages such as English and Chinese are relatively complete, but they are not perfect for more difficult-to-recognize texts such as Arabic and Mongolian. Some current scene text detection algorithms can detect text in multiple languagesand even detect text in multiple languagesin the same scene. However, in a more complex and diverse language environment, today's scene text detection cannot be accurate and efficient. The goal. The main purpose of this part of the research is to achieve detection in complex and diverse language and text environments. It is necessary to study most language types and scene text detection methods, approaching an algorithm that is most universal in various languages, and based on deep learning. The language dataset enables the scene text detection network to analyze and detect the input text of various language types.

In fact, in order to obtain a scene text detection model that can detect multiple languages, complex structures, huge networks and many modules are necessary. However, the huge computation and numerous modules and connections will increase the running inference time, which is not conducive to the short-time consumption in practical applications. Therefore, we need to find a balance between the two and continuously optimize to make text detection fast, accurate and universal.

5. Conclusion

Text detection and recognition in natural scenes is currently one of the research hotspots in the field of computer vision and pattern recognition, which is of great significance for obtaining information from scenes. This paper mainly sorts out, analyzes and summarizes the scene text detection method and its development based on deep learning. Firstly, the related research background and significance of scene text detection are discussed. Then, the second part expounds some main technical research routes of scene text detection. According to the timeline of detection methods, the specific contents of various text detection models are further introduced. Third, this paper compares and analyzes the experimental results of different models. In addition, some model improvements are further introduced, including relationships, effects, strengths and weaknesses, and expectations. Finally, the challenges and development trends of scene text detection technology based on deep learning are summarized.

References

[1] Zhi Tian, Weilin Huang, Tong He, Pan He, Yu Qiao. 2016. Detecting Text in Natural Image with Connectionist Text Proposal Network. ECCV, 2016. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1609.03605

[2] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, Jiajun Liang. 2017. EAST: An Efficient and Accurate Scene Text Detector. CVPR 2017. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1704.03155

[3] Baoguang Shi, Xiang Bai, Serge Belongie. 2017. Detecting Oriented Text in Natural Images by Linking Segments. CVPR 2017. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1703.06520

[4] Minghui Liao, Baoguang Shi, Xiang Bai, Xinggang Wang, Wenyu Liu. 2017. TextBoxes: A Fast Text Detector with a Single Deep Neural Network. AAAI2017. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1611.06779

[5] Minghui Liao, Baoguang Shi, Xiang Bai. 2018. TextBoxes++: A Single-Shot Oriented Scene Text Detector. IEEE Transactions on Image Processing 27 (2018). Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1801.02765

[6] Wenhai Wang, Enze Xie, Xiang Li, Wenbo Hou, Tong Lu, Gang Yu, Shuai Shao. 2019. Shape Robust Text Detection with Progressive Scale Expansion Network. CVPR 2019. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1903.12473

[7] Chuhui Xue, Shijian Lu, Wei Zhang. 2019. MSR: Multi-Scale Shape Regression for Scene Text Detection. IJCAI19. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1901.02596

[8] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, Chunhua Shen. 2019. Efficient and Accurate Arbitrary-Shaped Text Detection with Pixel Aggregation Network. ICCV 2019. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/1908.05900v1

[9] Yuliang Liu, Hao Chen, Chunhua Shen, Tong He, Lianwen Jin, Liangwei Wang. 2020. ABCNet: Real-time Scene Text Spotting with Adaptive Bezier-Curve Network. CVPR 2020. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/2002.10200v2

[10] Yiqin Zhu, Jianyong Chen, Lingyu Liang, Zhanghui Kuang, Lianwen Jin, Wayne Zhang. 2021. Fourier Contour Embedding for Arbitrary-Shaped Text Detection. CVPR 2021. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/2104.10442

[11] Mingxin Huang, Yuliang Liu, Zhenghao Peng, Chongyu Liu, Dahua Lin, Shenggao Zhu, Nicholas Yuan, Kai Ding, Lianwen Jin. 2022. SwinTextSpotter: Scene Text Spotting via Better Synergy between Text Detection and Text Recognition. CVPR 2022. Computer Vision and Pattern Recognition (cs.CV). Address of the paper: https://arxiv.org/abs/2203.10209