Computers, Materials & Continua CMC, vol.65, no.3, pp.2233-2245, 2020

A Modified Method for Scene Text Detection by ResNet

Shaozhang Niu[1, *], Xiangxiang Li[1], Maosen Wang[1] and Yueying Li[2]

Abstract: In recent years, images have played a more and more important role in our daily life and social communication. To some extent, the textual information contained in the pictures is an important factor in understanding the content of the scenes themselves. The more accurate the text detection of the natural scenes is, the more accurate our semantic understanding of the images will be. Thus, scene text detection has also become the hot spot in the domain of computer vision. In this paper, we have presented a modified text detection network which is based on further research and improvement of Connectionist Text Proposal Network (CTPN) proposed by previous researchers. To extract deeper features that are less affected by different images, we use Residual Network (ResNet) to replace Visual Geometry Group Network (VGGNet) which is used in the original network. Meanwhile, to enhance the robustness of the models to multiple languages, we use the datasets for training from multi-lingual scene text detection and script identification datasets (MLT) of 2017 International Conference on Document Analysis and Recognition (ICDAR2017). And apart from that, the attention mechanism is used to get more reasonable weight distribution. We found the proposed models achieve 0.91 F1-score on ICDAR2011 test, better than CTPN trained on the same datasets by about 5%.

Keywords: CTPN, scene text detection, ResNet, attention.

1 Introduction

Scene text detection technology in computer vision has become the research focus for many years, because of its high application value in reality. For instance, researchers have used the technology of scene text detection and recognition to work out a complete available system for automatic license plate recognition in unconstrained capture scenarios [Montazzolli and Rosito (2018)]. With the continuous development of intelligent terminals, there have appeared more and more instant mobile photo translation and recognition applications, which are also based on the text detection of the captured scenes. However, the results of scene text detection are often not so ideal, due to the enormous differences between the

[1] School of Computer Science and Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China.

[2] College of Arts and Sciences, Boston University, Boston University, Boston, 02215, USA.

[*] Corresponding Author: Shaozhang Niu. Email: szniu@bupt.edu.cn.

Received: 18 December 2019; Accepted: 21 June 2020.

scenes themselves, the high complexity of the scene background, and the various light intensity. So, it is still a challenging task to get more accurate results. Inspired by the Region Proposal Network (RPN) presented in Faster Region-based Convolutional Neural Network (Faster-RCNN) [Ren, He, Girshick et al. (2015)], Tian et al. [Tian, Huang, He et al. (2016)] proposed CTPN using vertical anchor mechanism, which detects text with fixed-width multi-size proposals. In this work, we also decide to utilize vertical anchor mechanism to detect text in natural scene images. But rather than VGGNet [Simonyan and Zisserman (2014)] used in CTPN, we use ResNet [He, Zhang, Ren et al. (2015)] to extract the features of the images. Because of the shortcut connection, ResNet can improve the performance in the process of deepening the number of network layers, thus having an obvious advantage compared with VGGNet. Different from the low-level features in previous models, the deeper convolutional neural network can provide more abstract features and stronger semantic information, which enable our neural network to be less affected by the differences between different images. And the number of parameters in ResNet is also fewer than that of VGGNet, due to the bottleneck design itself. Specifically, compared with ResNet-152, the number of weight parameters for ResNet-50 and ResNet-101 is within acceptable range while having almost the same performance, according to the data in ResNet. So, we decided to use ResNet-50 and ResNet-101 to implement the specific experiments. We also use the attention mechanism to get more accurate weight distribution of context sequence information. We train our models with the datasets of MLT from ICDAR2017 competition, which is beneficial to enhancing the robustness of our trained models for multiple languages. And the number of images in our datasets is obviously larger than that of CTPN, which could make the extracted image features more diverse, thus, to some extent, beneficial to improving the text detection ability of the models we trained.

2 Related work

2.1 Object detection

Region-based Convolutional Neural Network (R-CNN) [Girshick, Donahue, Darrelland et al. (2014)] is a classic model for object detection in deep learning. The model combines the features extracted by convolutional neural network and the region proposals generated by Selective Search [Uijlings, Van de sande, Gevers et al. (2013)]. Its performance on the PASCAL Visual Object Classes Challenge 2012 (PASCAL VOC 2012) datasets improved by 30%, compared to the state-of-the-art models at that time, and in so promotes the emergence of subsequent Fast Region-based Convolutional Neural Network (Fast-RCNN) [Girshick (2015)] and Faster-RCNN. Updated from Fast-RCNN, rather than the Selective Search used before, Faster-RCNN uses RPN to generate proposals, and the other parts such as feature extraction and classification also follow the usage in Fast-RCNN. So we can approximately regard Faster-RCNN as the combination of RPN and Fast-RCNN. Specifically, Faster-RCNN extracts the features of the images by using VGGNet. This model generates proposals by sliding a small 3×3 window on the feature map, and finally classifies these proposals. But according to Ren et al. [Ren, He, Girshick et al. (2015)], the processing speed of Faster-RCNN can only reach 17 fps even with the smaller Zeiler and Fergus (ZF) model [Zeiler and Fergus (2014)] to extract image features, which is still somewhat slow compared with 30fps required by real-time

systems. However, the problem has been solved with the appearance of Single Shot Detector (SSD) [Liu, Anguelov, Erhanet et al. (2016)]. SSD is close to Faster-RCNN in precision while much faster in the speed of object detection. The main contribution of SSD lies in its use of multilayer network features. The situation has then changed a lot since the emergence of ResNet. ResNet has solved the problem of vanishing gradient in the processing of back-propagation by means of the shortcut connection design, and therefore has enabled the convolution neural network to achieve continuous performance improvement when deepening the number of network layers. Meanwhile, this gave ResNet the first place in some subprojects of ImageNet Large Scale Visual Recognition Challenge 2015 (ILSVRC2015). Moreover, the original version of Faster-RCNN has also taken on new developments due to the appearance of ResNet. The researchers presented Region-based Fully Convolutional Network (R-FCN) [Dai, Li, He et al. (2016)], which has changed VGGNet to ResNet for image feature extraction, because of its deeper network layers and stronger ability to get semantic information. R-FCN is a region-based fully convolutional network for object detection. Because of the share of computing, the number of the weight parameters is smaller. All of the above make the test results improve distinctly in the field of precision and speed. For instance, it is mentioned in R-FCN that the test speed of the models is 170 ms/image, 2.5 to 20 times faster than Faster-RCNN. Mask Region-based Convolutional Neural Network (Mask R-CNN) [He, Gkioxari, Dollár et al. (2017)] is an extended object detection framework based on Faster-RCNN. Besides using ResNet and Feature Pyramid Network (FPN) for image feature extraction, it also utilizes the Region of Interest (ROI) align to replace the ROI pooling in Faster-RCNN. Mask R-CNN has a strong robustness in the field of human posture estimation, which has a profound impact on academia and industry.

2.2 Scene text detection

To some degree, scene text detection can be taken as a special case of object detection, while there are still some differences between them. Generally, object detection is required to find out the location of the targets in the input images and give out their respective categories, while scene text detection needs to figure out the text in the input images with the precision of word-level or text-line-level. As such, the precision of scene text detection is much higher than that of object detection. In addition, the text lines or words to be detected are often a sequence that is composed of one or multiple characters, which may present a large difference or a long distance, therefore it is obviously more difficult than objects detection. In summary, the general approaches of object detection cannot be directly utilized in scene text detection. Consequently, we should make some improvements for specific text detection situations.

In recent years, plenty of scholars and researchers have contributed to the field of scene text detection in computer vision. Efficient and Accuracy Scene Text (EAST) [Zhou, Yao, Wen et al. (2017)] can produce detection results of word-level or text-line-level directly with only two stages. By sending the first stage prediction to Non-Maximum Suppression (NMS), we can get the final F1-score of 0.7820 (0.8072 when using multi-scale) in ICDAR 2015. Segment Linking (Seglink) [Shi, Bai and Belongie (2017)] first detects the text using a combination of segment and link, where the segment part can detect a part of words or text lines and the link part connects those adjacent segments that

belong to the same words or text lines, then the final detection results come into being. Different from methods such as Seglink in Deng et al. [Deng, Liu, Li et al. (2018)], a new type of method of scene text detection called Pixel Linking (PixelLink) was proposed by researchers based on instance segmentation. The model is inspired by SegLink, but has achieved the link of pixel-level. This network doesn't use the bounding box regression that is employed in many other preceding algorithms. However, while getting the comparable results with less training data and iterations, it is a great breakthrough in scene text detection. Lyu et al. [Lyu, Yao, Wu et al. (2018)] has proposed the model that could detect multi-oriented scene text by using corner localization and region segmentation while averting their weakness. With VGGNet to extract features, it has achieved the F1-score of 0.843 on ICDAR2015 and 0.815 on MSRA Text Detection 500 Database (MSRA-TD500). TextBoxes [Liao, Shi, Bai et al. (2017)] is an end-to-end scene text detector utilizing fully convolutional network. The method improves the aspect ratio of the default boxes based on SSD and introduces a relatively long kernel to adapt the large aspect ratio in scene text, thus finally achieving the F1-score of 0.86 in ICDAR2013. As an extension of the TextBoxes, Textboxes++ [Liao, Shi and Bai (2018)] has developed the scope of text detection from horizontal to arbitrary orientations, which enables the network to be more robust in detection scenes. Furthermore, the TextBoxes series also have a scene text recognition part based on Convolutional Recurrent Neural Network (CRNN) [Shi, Bai and Yao (2015)] after the detection. To some degree, the integration of detection and recognition is also a highlight of TextBoxes and TextBoxes++. Supervised Pyramid Context Network (SPCNET) [Xie, Zang, Shao et al. (2019)] is an effective scene text detection model based on FPN and instance segmentation. This model is mainly inspired by Mask R-CNN and could detect curved text in the real world with top performance at present.

2.3 Attention mechanism

When we notice a specific scene, we have different attention distribution to each part of it, and when we look at somewhere else, our attention shifts as our eyes move. There is also something similar to this in our daily life. The phenomenon is called attention mechanism and is frequently discussed, not only in academia, but also, recently, in the industry. To some extent, we can divide attention mechanism into soft attention and hard attention according to the format of its output vectors. The hard attention has the output of one-hot vector, and the soft attention weights each part in the input vectors of attention by learning the parameters of each element from back-propagation. Moreover, we can also divide attention into spatial attention and channel attention. The spatial attention gives different weights to different space feature regions [Woo, Park, Lee et al. (2018)], and the channel attention gives various weights to features of various channels [Hu, Shen, Sun et al. (2017)]. Essentially, the purpose of attention is to learn a weight distribution, and then apply it to the input feature vectors of attention. It should be noted that the learned weight distribution can be either applied to the original input images or the convolution feature map.

3 Our proposed network

As mentioned before, our newly proposed network replaces the VGG16 used in CTPN with the more powerful ResNet. The specific process of network replacing is as follows. It is implied in the previous reference Simonyan et al. [Simonyan and Zisserman (2014)] that the conv5_3 of VGG16 has the feature map of 14×14 and the stride of 16. So, when we slide a small window in conv5_3 with the stride of 1, we actually pass through 16 pixels on the original input images, which corresponds well to the fixed width of the anchors (16 pixels) set in CTPN. The ResNet, which is diverse from VGG16, can be divided into conv1, conv2_x, conv3_x, conv4_x and conv5_x. According to the detailed data in He et al. [He, Zhang, Ren et al. (2015)], the size of feature map in conv4_x is 14×14, and its stride is 16, which is consistent with that of VGG16. This means that in principle, we can replace VGG16, used in CTPN, with ResNet and implement further experiments without changing the preceding fixed width of the anchors. So, we finally use the convolutional part of ResNet (from input to conv4_x) to take the place of VGG16 (from input to conv5_3). According to the performance of ResNet in He et al. [He, Zhang, Ren et al. (2015); Dai, Li, He et al. (2016)], it is believed that the newly proposed network should also be able to get a better result for scene text detection.

The structure of our network is shown in Fig. 1 (using ResNet101). First, the training images are sent to ResNet for feature extraction. To get the anchors, we slide a 3×3 window through the res4b22 layer of conv4_x. It should be noted that each time, a window sliding can generate various sizes of anchors, whose generation mechanism is listed as follows. First, the width of the anchors is fixed at 16 pixels (the size in the input images) in the horizontal direction. Then, to fit the diverse sizes of text in actual scenes, the height of the anchors varies between 11-273 (each time divided by 0.7, a total of 10 size changes) in the vertical direction. In additional, we use the Long Short Term Memory (LSTM) to fully utilize the contextual association between isolated characters in images. LSTM is a special case of Recurrent Neural Network (RNN). Furthermore, there is an important feature of RNN in that it can use the input of current moment and the output of the layer at previous moment as the whole input of current moment, which is beneficial for the network to make a comprehensive judgment by using some information before. But sometimes, to decide the output of current moment, some information in the future may be used as an aid to make a more accurate judgment. For example, if there are two or more interpretations of the words currently to be recognized in some speech recognition tasks, it may be necessary to judge the output of current content according to the voice information input to the network afterwards. Here is the exact time for us to use Bidirectional RNN (BRNN). We all know that one of the major flaws of RNN is the long-term dependency problem, which makes it impossible to synthesize information far from the current moment. However, by introducing the input gate, output gate, forget gate and the special structural combination between them, LSTM has solved the problem skillfully. Therefore, the BLSTM is applied to the network to connect the isolated proposals from the context information. Apart from that, we also use the attention mechanism to get more precise context sequence information by applying it between the BLSTM and the FC layer, so that the output of BLSTM could learn a more reasonable weight distribution. In addition, we can also better predict the content of current 16 pixels proposals (text or not text). This operation has improved the final scene text detection effect.

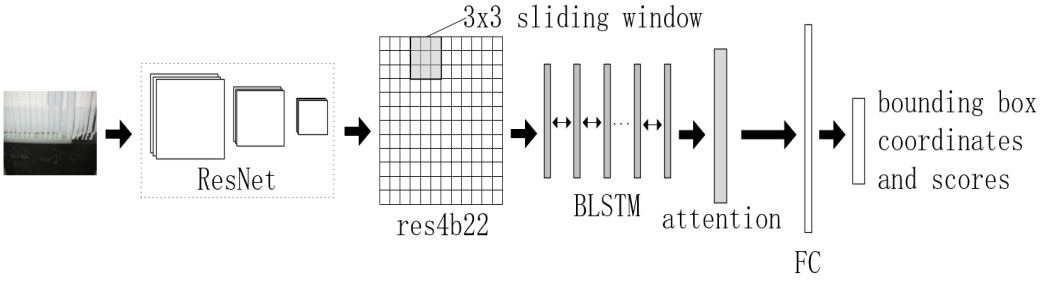

Figure 1: The structure of the network

Besides, there are some other requirements when connecting and merging these predicted proposals: firstly, the merged proposals should have a text/non-text score greater than 0.9. Then, the two adjacent proposals can be merged when their distance is less than 50 pixels. Finally, the overlap rate between two adjacent proposals in the vertical direction should be greater than 0.7. As shown in Fig. 2, we can see the detection results of the input images before and after the merging. The upper part of Fig. 2 shows the results before merging, while the lower part shows the results after merging.

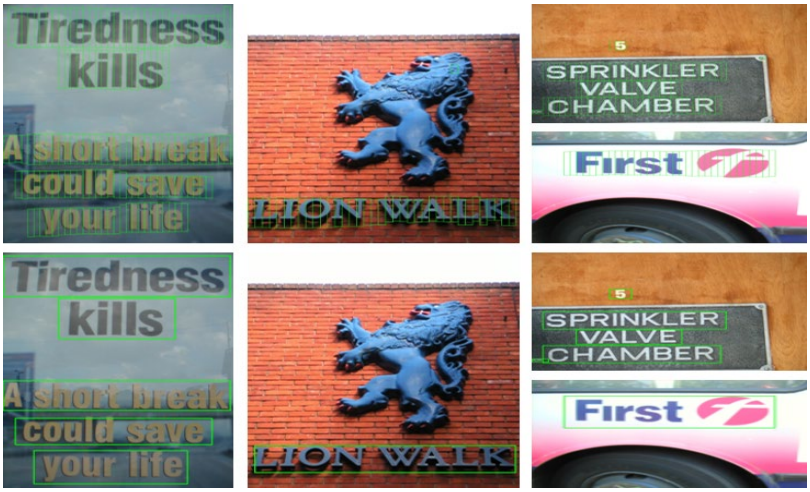

Figure 2: The detection results of the input images before and after the merging

4 Experiments

4.1 Experiments about text localization

We have evaluated the test results of our proposed network by the benchmark of ICDAR2011 (Born-Digital Images (Web and Email)) and ICDAR2013 (Task 2.1: Text Localization). The following is an example to illustrate the test datasets and test process of ICDAR2013. The datasets downloaded from the official website include 233 natural images with text information to be detected. In order to better test the detection effect of

our models under different scenarios (as shown in Fig. 3), the clarity, the lighting conditions of background and the size of the text in images are chosen to be all different from each other. We used the trained models to generate predicted proposals on these test images and saved them with visible information to observe the specific detection effect. Additionally, we have saved the coordinates with the format of (Xmin, Ymin, Xmax, Ymax). The four straight lines corresponding to the above x, y coordinates can be enclosed in a rectangle on the coordinate axis, which is exactly the final proposals predicted by our models. When we save the coordinates of the proposals, a proposal corresponds to one line in the above format, and a file corresponds to all coordinates on an image. In other words, how many proposals there are predicted on an image, how many lines there are in a file. Finally, the predicted 233 files with coordinate information are packaged into a zip format file and upload to the official website to verify the test results of our models.

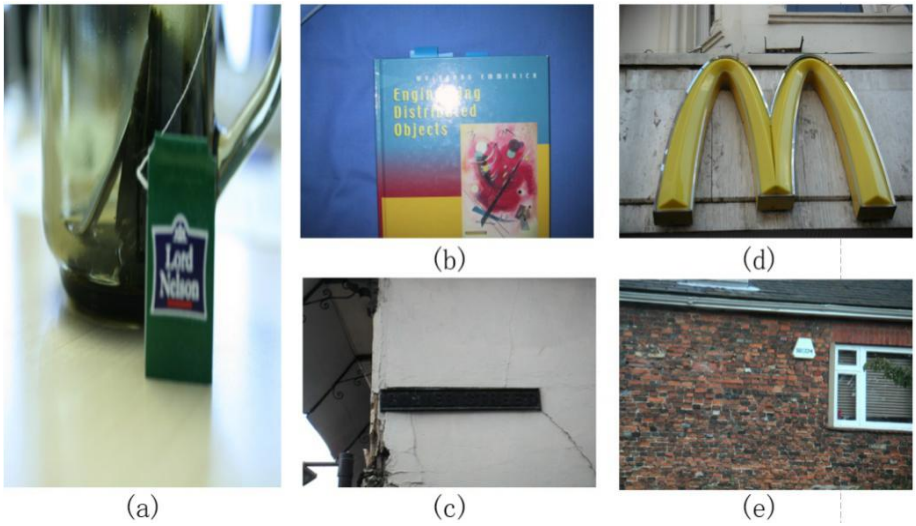

Figure 3: Images of extreme scene

We can learn from Fig. 3 that: Fig. 3(a) represents the fact that the text information to be detected is very vague, Fig. 3(b) represents the case where the text information to be detected is illuminated by strong light, Fig. 3(c) represents the extremely dim background of the text to be detected, Fig. 3(d) represents the case where the text in the scenes is large, and Fig. 3(e) represents the case where the text in the scenes is very small. In general, Fig. 3(b) is contrasted with Fig. 3(c), and Fig. 3(d) is contrasted with Fig. 3(e). We have detected Fig. 3(a), Fig. 3(e) with CTPN and with our modified network. The detection results are shown in Fig. 4, and the upper part is the detection results of our models. We can learn from Fig. 4 that both models have detected 1 proposal while Fig. 4(a) has two text lines, and our models have detected 1 proposal of the small text in Fig. 4(e). In summary, the overall performance gap between two models is not large, but the detection of extreme natural scenes is still a challenge. Compared with the images in ordinary natural scenes, we can distinguish the text from the background more difficult.

On the other hand, the number of extreme scene image samples are not as many as that of common scene images. Therefore, we have less training for complex natural scenes, which results in the poor performance when detecting text in these situations. This is a direction of our future efforts.

Figure 4: Detection results of picture (a) and picture (e)

The test results of ICDAR2011 are shown in Tab. 1. The "Model" column represents the different models proposed in different papers to be compared. Then, the "P", "R", and "F" in the header represent precision, recall, and F1-score respectively. The value of F1-score combines the values of precision and recall, which is a comprehensive judgment to test the models. Our (ResNet101) and our (ResNet50) in Tab. 1 are the experimental data of our newly proposed network, and the CTPN* represents the results of training CTPN under the same datasets as our new network. It can be seen from Tab. 1 that CTPN* and our (ResNet50) attained the F1-score of 0.86 and 0.88 respectively. When replacing VGG16 with ResNet50, the result has improved by 2%, and when we use the deeper ResNet101 for training, the result is 0.91, improving by 3% compared to ResNet50, 5% compared to CTPN* trained on the same datasets. Moreover, we have tested our models on ICDAR2013 benchmark, which is shown in Tab. 2. Under the premise of the same training datasets for CTPN*, ResNet50 and ResNet101, the data in Tab. 2 also shows an experimental phenomenon similar to that in Tab. 1. In a word, the test results of CTPN*, our (ResNet50) and our (ResNet101) on ICDAR2011 and ICDAR2013 should be able to explain that: compared with VGG16, ResNet has stronger ability for image feature extraction, thus getting better results on scene text detection.

Table 1: Test results of ICDAR2011

Model	P	R	F
TextFlow [Tian, Pan, Huang et al. (2015)]	0.86	0.76	0.81
Text-CNN [He, Huang, Qiao et al. (2015)]	0.91	0.74	0.82
USTB_TexStar	0.94	0.87	0.90
CTPN*	0.81	0.92	0.86
our (ResNet50)	0.85	0.90	0.88
our (ResNet101)	0.91	0.90	0.91

Table 2: Test results of ICDAR2013

Model	P	R	F
SegLink	0.87	0.83	0.85
Fastext [He, Huang, Qiao et al. (2015)]	0.84	0.69	0.77
Text-CNN	0.93	0.73	0.82
Multi-Oriented-FCN	0.88	0.78	0.83
TextBoxes	0.89	0.83	0.86
BayesText	0.85	0.67	0.75
our (ResNet50)	0.81	0.80	0.80
CTPN*	0.78	0.81	0.79
our (ResNet101)	0.87	0.78	0.82

As shown in Tab. 3, to implement another group of experiments, we utilize ResNet101 as the backbone to extract the features of images. The difference is that we have used attention this time. We have carried out our experiments using the test datasets of ICDAR2013, we also tested the average processing speed of these 233 images, and the final results of comparison are shown below. It is indicated that the F1-score has improved by 1% with the use of attention. On the other hand, due to the additional steps compared to previous network, the average processing speed of images has also slowed down accordingly, as is expected.

Table 3: Test results of ICDAR2013 (using backbone of ResNet101)

Network	P	R	F	Average Speed (s)
No attention	0.87	0.78	0.82	0.167
With attention	0.86	0.80	0.83	0.183

We finally use 6000 images downloaded from official website of ICDAR2017 Competition. It should be noted the labels given in the datasets are word-level. But from the network architecture presented in Part 3 and the description of the experiments in Part 4, we are able to know that the experiments use the vertical anchor mechanism proposed in CTPN to fix the width of the anchors to 16 pixels. Therefore, the labels should also be scaled to 16 pixels in advance before training. So, our experimental work also includes converting the word-level labels to fixed-width 16 pixels labels. And we regard it as 16 pixels when the width is less than 16 pixels.

4.2 Experiments by post-processing

Because of the limitations of the network itself, the original CTPN can only detect text in horizontal direction. However, by doing some transformation and fitting to the proposals predicted by the network, we can detect the text information in images with less inclination. As shown in Fig. 5, when the text is not in the horizontal direction, we fit a slanted line according to the center point of the adjacent proposals. The upper and the lower borders of the merged rectangles are parallel to the lines we got, and the height after merged is related to the average height of all proposals. Then the coordinates of the four corners for the merged rectangles are related to the maximum and minimum x, y coordinates of the top and bottom borders of all proposals. Although the treatment is slightly backward compared with the current cutting-edge methods, it still has significance as an improvement to CTPN in some degree.

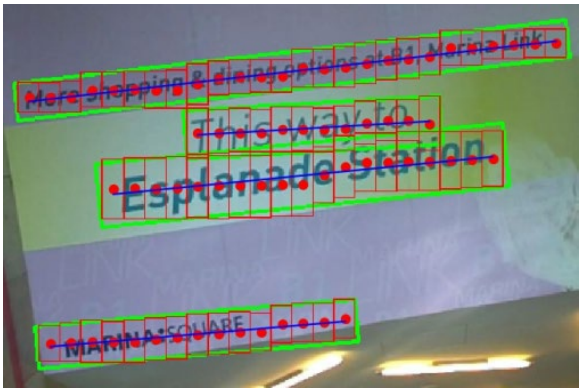

Figure 5: The detect results of multi-orientation text with small tilt

As part of our work, we expanded our models to text recognition by post-processing on the current basis. First, we extracted the merged predicted proposals from the input images according to the corresponding coordinates. Then, we sent these coordinates to a scene text recognition network CRNN, which utilizes Connectionist Temporal Classification (CTC) to get the final predicted sequences. The advantage of CTC is that we don't need to know in advance how many characters are needed to be identified in the predicted sequences. CTC mainly solves the problem of label alignment, and the principle of it can be described as follows: we use a sequence $y_1, y_2, \ldots, y_t$ as the input, where "t" is the length of the sequence. Thus, y_t^{π} means that we get "π" as the output of time stamp "t". If we use p (π | x) represents the probability that the input is "x" and the output is "π", since the output of every time stamp is independent of each other, the p (π | x) can be expressed as follows:

$$p(\pi|x) = \prod_{t=1}^{T}(y_t^{\pi}) \quad (1)$$

Then we define a function "F" maps from "π" to the label sequence "l" by removing the repeated labels and the blanks. For instance, by means of mapping function F, we can map "ss-c--h-o-o-l" to "school" naturally ("-" represents a blank), so the p (l | x) can be represented as follows:

$$p(l|x) = \sum_{\pi \in F^{-1}(l)} p(\pi|x) \quad (2)$$

And finally, we can get the most possible output sequence "l*" by the following formulation (3):

$$l^* = \underset{l}{\operatorname{argmax}} p(l|x) \quad (3)$$

We display the results identified by CRNN in the upper left corner of the detected rectangles (as shown in Fig. 6). It can be found that when the conditions of light, background and other conditions are relatively good, we can see a certain recognition effect.

Figure 6: The recognition results of text by post-processing

5 Conclusion

In this paper, we presented a modified scene text detection network. To extract stronger semantic information, our models replace VGG16 in CTPN with deeper ResNet. At the same time, we use attention mechanism to get more accurate context information. In addition, the datasets of MLT from ICDAR2017 are used to improve the robustness of our models to multiple languages. At present, the models can only detect scene text with horizontal or small tilt. However, we get unsatisfactory results on vague text or characters with large font. Although we initially have combined the scene text detection with recognition, this is not really an end-to-end network. Moreover, the current recognition results need to be further optimized and improved. As a conclusion, we may solve these problems gradually in our future work, and we will try to develop scene text detection to arbitrary orientations, or establish a real end-to-end scene text detection and recognition network based on the improvement of detection accuracy.

Acknowledgement: The authors thank Dr. Jiancheng Zou at North China University of Technology for his helpful advice. The authors are also grateful to the very thorough reviewers.

Funding Statement: This work was supported by National Natural Science Foundation of China (Nos. U1536121, 61370195).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Dai, J.; Li, Y.; He, K.; Sun, J. (2016): R-FCN: object detection via region-based fully convolutional networks. *Advances in Neural Information Processing Systems*, pp. 379-387.

Deng, D.; Liu, H.; Li, X.; Cai, D. (2018): Pixellink: detecting scene text via instance segmentation. *Association for the Advancement of Artificial Intelligence*, pp. 6773-6780.

Girshick, R. (2015): Fast r-CNN. *IEEE International Conference on Computer Vision*, pp. 1440-1448.

Girshick, R.; Donahue, J.; Darrelland, T.; Malik, J. (2014): Rich feature hierarchies for accurate object detection and semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587.

He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. (2017). Mask r-CNN. *IEEE Conference on Computer Vision*, pp. 2961-2969.

He, K.; Zhang, X.; Ren, S.; Sun, J. (2015): Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778.

He, T.; Huang, W.; Qiao, Y.; Yao, J. (2015): Text-attentional convolutional neural network for scene text detection. *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2529-2541.

He, T.; Huang, W.; Qiao, Y.; Yao, J. (2015): Fastext: efficient unconstrained scene text detector. *IEEE International Conference on Computer Vision*, pp. 1206-1214.

Hu, J.; Shen, L.; Sun, G. (2017): Squeeze-and-excitation networks. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132-7141.

Liao, M.; Shi, B.; Bai, X.; Wang, X.; Liu, W. (2017): Textboxes: a fast text detector with a single deep neural network. *Association for the Advancement of Artificial Intelligence*, pp. 4161-4167.

Liao, M.; Shi, B.; Bai, X. (2018): Textboxes++: a single-shot oriented scene text detector. *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3676-3690.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S. et al. (2016): SSD: single shot multibox detector. *European Conference on Computer Vision*, pp. 7310-7319.

Lyu, P.; Yao, C.; Wu, W.; Yan, S.; Bai, X. (2018): Multi-oriented scene text detection via corner localization and region segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7553-7563.

Montazzolli, S.; Rosito, C. (2018). License plate detection and recognition in unconstrained scenarios. *European Conference on Computer Vision*, pp. 580-596.

Ren, S.; He, K.; Girshick, R.; Sun, J. (2015): Faster r-CNN: towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, pp. 91-99.

Simonyan, K.; Zisserman, A. (2015): Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*, pp. 1-14.

Shi, B.; Bai, X.; Belongie, S. (2017): Detecting oriented text in natural images by linking segments. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2550-2558.

Shi, B.; Bai, X.; Yao, C. (2015): An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 11, pp. 2298-2304.

Tian, S.; Pan, Y.; Huang, C.; Lu, S.; Yu, K. et al. (2015): Text flow: a unified text detection system in natural scene images. *IEEE International Conference on Computer Vision*, pp. 4651-4659.

Tian, Z.; Huang, W.; He, T.; He, P.; Qiao, Y. (2016): Detecting text in natural image with connectionist text proposal network. *European Conference on Computer Vision*, pp. 56-72.

Uijlings, J. R.; Van De Sande, K. E.; Gevers, T.; Smeulders, A. W. (2013): Selective search for object recognition. *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154-171.

Woo, S.; Park, J.; Lee, J. Y.; Kweon, I. S. (2018): CBAM: convolutional block attention module. *European Conference on Computer Vision*, pp. 3-19.

Xie, E.; Zang, Y.; Shao, S.; Yu, G.; Yao, C. et al. (2019). Scene text detection with supervised pyramid context network. *Association for the Advancement of Artificial Intelligence*, pp. 9038-9045.

Zeiler, M. D.; Fergus, R. (2014): Visualizing and understanding convolutional networks. *European Conference on Computer Vision*, pp. 818-833.

Zhou, X.; Yao, C.; Wen, H.; Wang, Y.; Zhou, S. et al. (2017): East: an efficient and accurate scene text detector. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5551-5560.