A scene text detector based on deep feature merging

Yong Zhang[1,2,3]

[1] ATR Key Laboratory of National Defense Technology, Shenzhen University, Shenzhen 518060, China

[2] Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China

[3] Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen University, Shenzhen 518060, China

Yubei Huang[1]

[1] ATR Key Laboratory of National Defense Technology, Shenzhen University, Shenzhen 518060, China

Donning Zhao[4]

[4] Shenzhen Vetose Technology Co. Ltd, Shenzhen 518102, China

Chun Ho Wu[5]

[5] Department of Supply Chain and Information Management, Hang Seng University of Hong Kong, Hong Kong 999077, China

Wai Hung Ip[6]

[6] Department of Industrial and Systems Engineering, the Hong Kong Polytechnic University, Hong Kong 999077, China

Kai Leung Yung[6]

[6] Department of Industrial and Systems Engineering, the Hong Kong Polytechnic University, Hong Kong 999077, China

Abstract
Scene text detection has become an important research topic. It can be broadly applied to much industrial equipment, such as smart phones, intelligent scanners, and IoT devices. Many existing scene text detection methods have achieved advanced performance. However, text in scene images is presented with differing orientations and varying shapes, rendering scene text detection a challenging task. This paper proposes a method for detecting texts in scene images. First, four stages of low-level features is extracted using DenseNet121. Low-level features are then merged by transposed convolution and skip connection. Second, the merged feature map is used to generate a score map, box map, and angle map. Finally, the Locality-Aware Non-Maximum Suppression (LANMS) is applied as post-processing to generate the final bounding box. The proposed method achieves an F-measure of 0.826 on ICDAR 2015 and 0.761 on MSRA-TD500, respectively.

Keywords Scene text detector · Deep feature merging · DenseNet121 · Convolutional neural network

Introduction

With the rapid development of deep learning, many computer vision tasks have achieved promising results in recent years, such as image classification [7], object detection [13, 19], crowd flows prediction [1] and scene text detection [2, 16, 18, 21, 24, 31]. Accurately predicting the location of text in natural scenes constitutes a fundamental step in developing advanced applications such as text recognition, scene analysis, and autopilot, therefore scene text detection has become one of the key research tasks in computer vision. However, due to some basic characteristics of scene text (such as differences in text orientation, diversity of shapes, and image blurring caused by inadequate imaging conditions), scene text detection remains a challenging task.

In addition to the applications mentioned above, scene text detection has also been applied in intelligent transportation, scene text detection can be combined with monitoring systems and vehicle tracking systems to detect the license plate numbers of illegal vehicles. In the area of intelligent logistics, automatic sorting robots use text detection techniques to detect text in packages, such as waybill numbers and destination addresses, and then combined with other sensors to plan operations. Equally, some IoT devices can be used in smart retail to detect bar codes and product names using scene text detection techniques. Scene text detection can also be used in other smart retail scenarios, such as with vending machines and self-service stores (e.g. Amazon Go).

In recent years, many scene text detection methods [3–6, 8, 9, 17, 22, 30] have achieved advanced results in some benchmark datasets. As with many other object detection tasks, the aim of scene text detection is to design and extract text features from natural scene images. Traditional scene text detection methods [4, 6, 8, 17] detect text using features that are manually designed. Methods [3, 5, 9, 30] based on deep learning extract features automatically using a multi-layer convolutional network. Currently, most traditional and deep learning methods mainly focus on the text that is presented in simple scene images. However, current algorithms cannot achieve satisfactory results for either single orientation text that occurs in complex background images or multiple orientation scene text.

To detect text that lies in complex background scenes, we propose a scene text detection method based on EAST [30]. The proposed network model includes three parts: a backbone network, a feature merger, and an output layer. We use DenseNet121 [7] as a backbone network to extract four stages of low-level features and then merge them by transposed convolution and skip connection. The merged feature map is used to generate three output feature maps, which are the score, box, and angle maps.

We evaluated our method using ICDAR 2015 [10] and MSRA-TD500 [25], and the result showed that our method can detect texts from natural scene images accurately, achieving an F-measure of 82.6% on ICDAR 2015 and 76.1% on MSRA-TD500. The main contributions of this paper are as follows:

- We propose a scene text detection method, which achieved promising results on both ICDAR 2015 and MSRA-TD500.
- The proposed method introduces DenseNet121 as backbone, which enhance the validity of the low-level image feature, and introduces Dice and GIoU losses as part of loss function to improve the text detection accuracy.

In the remainder of this paper, we describe existing research on scene text detection in Section 2. Section 3 mainly focuses on the design of our proposed network model. Section 4 discusses loss functions. Experimental results are provided in Section 5, and we present our conclusion in Section 6.

Research on scene text detection

General object detection and scene text detection have become popular research topics recently, and many methods [13–14; 7–9] have achieved promising performances in the past few years. Zhu et al. [32] and Ye et al. [27] provide detailed reviews of the development of scene text detection and its current situation. In this section, we briefly review several scene text detection methods.

Traditional scene text detection methods mainly detect text using manually designed features. Based on the inherent feature of scene text, Maximally Stable Extremal Regions (MSER) [17] and Stroke Width Transform (SWT) [4] predicted scene text location by extracting single characters and then constructing words or text lines by grouping each adjacent character. Jaderberg et al. [8] used a multi-scale sliding window over the input image to generate many subareas and predict a text score for every sliding window. However, due to the complexity of the background of natural scenes and the diversity of the text, these algorithms are not sufficiently robust. For example, with the algorithm based on MSER, it is difficult to accurately extract the candidate text region when the text is situated in a complex background.

In recent years, scene text detection methods based on convolutional neural networks have achieved advanced performance. Huang et al. [6] extracted low-level text features using MSER and predicted text regions using a trained convolutional network classifier. CTPN [22] extracts fixed-width image feature maps as the input of a BLSTM and then uses the BLSTM to detect the connection between each character. Based on Faster R-CNN [19], R2CNN changes the scale and angle of ROI pooling modules to detect multi-orientation texts. SSTD [5] integrates a Text Attention Module (TAM) into Single Shot Multi-Box Detection (SSD) [13]. TAM was able to aggregate multi-level feature maps generated by SSD, which is the key to predicting text location. EAST [30] focused on the detection speed. It proposed a network model that was constructed by an FCN to predict shrink-text score maps and text location in performing a per-pixel regression and used Locality-Aware Non-Maximum Suppression (LANMS) as a post-processing method.

In this paper, a natural scene text detection method based on EAST [30] is proposed. The method offers several principal improvements. DenseNet121 is used as the backbone network, which can reduce the influence of gradient vanishing and model degradation, and improve the validity of basic image features. Transposed convolution is used as the up-sample method at the feature merging stage, which can enlarge the feature map and retain more image features than other methods. To improve the detection performance, text classification loss and bounding box loss in loss functions are redesigned. Experiments on ICDAR 2015 show that the proposed method outperforms EAST in terms of both precision and recall.

Network design

We proposed a model based on EAST [30] that was constructed through a convolutional neural network. Following EAST’s network design, our model consists of three parts: feature extraction, feature merging and output layering. Instead of using PVA-Net [11], DenseNet121 is used as backbone network. At the feature merging stage, we use transposed convolution to enlarge the feature map. Figure 1 is the overview of our proposed network model.

Feature extraction

The feature extraction branch represents the extraction of the basic feature maps of the image. Due to the complex background of the text in the natural scene, the robustness of the basic feature maps extracted through the backbone network has a crucial influence on the final result of the text detection. Therefore, the selection of the backbone network used in the feature extraction is particularly important. To reduce the impact of gradient vanishing and model degradation, we use DenseNet121 as the backbone network. Differing from other deep network models, DenseNet121 [7] reduces the number of network model parameters and alleviates the problems of gradient vanishing and model degradation by setting a network bypass to reuse the features. As we can see in Fig. 1, we use DenseNet121 to extract four stages of the convolutional feature maps from every input image.

Feature merging

The feature merging is mainly composed of up-sampling and feature maps concatenation along the channel axis. Unlike the feature merging used in EAST, we opted for transposed convolution as the up-sample method. The way our model merges features is as follows:

$$g_i = \begin{cases} TransposeConv(h_i) \; if \; i \leq 3 \\ conv_{3\times3}(h_i) \; if \; i = 4 \end{cases} \tag{1}$$

$$h_i = conv_{3\times3}(conv_{1\times1}([g_{i-1}; f_i])) \tag{2}$$

TransposeConv() means the transposed convolution, which used to enlarge the feature maps. g_i is the feature map that has been enlarged, and h_i is the merge feature map.

Output layering

The final output of the network model is related to the way the training data is generated. Following EAST's RBOX [30] geometry, the final output layer in our model contains 3 feature maps: a score map used to predict the probability of the text region, a text box map, and a text angle map to predict the bounding box.

Loss function

The loss of our method is formulated as:

$$loss = l_s + \lambda(l_b + l_a) \tag{3}$$

l_s represents the loss for the score map, and l_b and l_a represent the losses for the box and angle maps, respectively. λ is a weight, which balancing the loss of text classification and bounding box, it can be any positive number. In our experiment, we set λ to 1, due to the prediction of text and position of box are equally important in text detection

Loss of score map

The score map is used to predict the probability of the text region, which is a binary classification. We use Dice loss [15] for the score map:

$$L_s = DiceLoss = 1 - DiceCoef \tag{4}$$

DiceCoef means Dice coefficient, which is a function that measures the similarity between two score maps. It can be calculated as follows:

$$DiceCoef = \frac{2\left|\widehat{Y} \cap Y^*\right|}{\left|\widehat{Y}\right| + \left|Y^*\right|} \tag{5}$$

$Y*$ is the ground truth generated previously by the score map, and Y_b is the predicted score map.

Loss of box map

EAST uses Intersection over Union (IoU) loss [28] for the box map, which has two disadvantages. First, when there is no overlap between the predicted and the ground-truth bounding boxes, the value of IoU is 0, so the network cannot be further optimized. Second, it cannot accurately reflect the overlap position between the predicted and the ground-truth bounding boxes. As shown in Fig. 2, three overlapping cases have the same IoU, but it is obvious that the result of Fig. 2a is more accurate as far as the predicted location is concerned.

Therefore, we use GIoU [20] loss for the box map. It can be formulated as follows:

(1) Generate the minimum cover bounding box using the predictions of the bounding and ground- truth bounding boxes. As shown in Fig. 3, the blue box is the ground-truth, and the red box is that which is predicted by the model. The dashed line is the minimum cover bounding box.

(2) IoU is calculated by:

$$IoU = \frac{|R_1 \cap R_2|}{|R_1 \cup R_2|} \tag{6}$$

R_1 and R_2 are the areas of the ground-truth and predicted boxes.

(3) GIoU is calculated by:

$$GIoU = IoU - \frac{|R_3 \backslash (R_1 \cup R_2)|}{R_3} \tag{7}$$

R_3 is the area of the minimum cover bounding box.

(4) Consequently, the loss of box map is computed as:

$$l_b = 1 - GIoU = 1 - \frac{|R_1 \cap R_2|}{|R_1 \cup R_2|} + \frac{|R_3 \backslash (R_1 \cup R_2)|}{R_3} \tag{8}$$

Loss of angle map

Following EAST, we use cosine loss for the angle map:

$$l_a = 1 - cos\left(\theta^* - \widehat{\theta}\right) \tag{9}$$

$\widehat{\theta}$ is the predicted angle and $\theta^*$is the ground-truth angle.

Experiment

We tested our proposed method on two benchmark datasets: ICDAR 2015 [10] and MSRA-TD500 [25], comparing the proposed method with some existing algorithms.

1.1 Benchmark datasets

ICDAR 2015 contains 1000 training samples and 500 test samples, which are used for multi-orientation text detection tasks. The format of the text bounding box in the ICDAR 2015 consists of a clockwise movement between the coordinates of the four corner points of the text, starting from the upper left corner. Where a text has less than three pixels in the image area and is unrecognizable, it is marked as ignored. The images in ICDAR 2015 were all taken on a walk by a photographer wearing Google Glasses, so the images are distorted, over-angled, and blurred due to their jittery out-of-focus quality.

MSRA-TD500 is a multi-orientation long text dataset, marking the text bounding box with the text line. The dataset contains a total of 500 pictures, including 300 training images and 200 test images. All images are taken by the camera, and most of the text in the images is not clearly distinguished from the background. The text languages are mainly Chinese and English. The bounding box is marked by center coordinates and height, width, and rotation radians of the text region.

Experimental protocols

Following the object detection task, the performance of a natural scene text detection algorithm is generally measured by three protocols: Precision, Recall, and the F-measure. Precision and Recall can be calculated as follows:

$$Precision = \frac{|T|}{|D|} \tag{10}$$

$$Recall = \frac{|T|}{|G|} \tag{11}$$

Precision represents the proportion of correct results among all test results, and Recall represents the ratio of correct test results to all ground truths. T is the number of correct test results, D is the number of all test results, and G is the number of ground truths in the test set. Finally, the F-measure is used to represent the overall performance of the algorithm by integrating Precision and Recall:

$$Fmeasure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{12}$$

Experiment setup

Our experiments were conducted with Python3 and PyTorch. During training, we used Adam

[12] for the optimization algorithm. The initial learning rate was 0.0001 and decayed 1/10 every 10,000 epochs. The number of training samples was 20 every batch. We trained our model on two Tesla P100 GPUs. LANMS [30] was used as the post-processing method to generate the final bounding box.

Table 1 Comparison with ICDAR 2015

Method	Precision	Recall	F-Measure
Tian et al. [22]	0.74	0.52	0.61
Zhu et al. [31]	0.81	0.91	0.85
Shi et al. [31]	0.731	0.768	0.75
Xu et al. [24]	0.843	0.805	0.824
Ma et al. [14]	0.822	0.732	0.774
Deng et al. [3]	0.887	0.786	0.833
Zhou et al. [30] (EAST)	0.806	0.713	0.757
Ours	0.846	0.807	0.826

Experiment results

We compared our experiment results on ICDAR 2015 against other published methods in Table 1. The results of our method and EAST are reported on single-scale test images. On ICDAR 2015, which is designed for multi-orientation text detection, our method achieves an F-measure of 0.826, outperforming EAST by 6.9%. Some of the detection results are shown in Fig. 4a. We further evaluate the proposed method on MSRA-TD500, which is a text line dataset. On MSRA-TD500, as we can see in Table 2, we achieve competitive performance with a Precision of 0.813, a Recall of 0.715, and an F-measure of 0.761. This shows that the ability of our method to detect long text lines is stronger than baseline, which is EAST, especially on Recall. But compared with Deng et al. [3] and Zhu et al. [31], our algorithm is not good enough in terms of precision and recall. Figure 4b displays some examples of detection comparing our method with that of MSRA-TD500.

Table 2 Comparison on MSRA-TD500

Method	Precision	Recall	F-Measure
Zhang et al. [29]	0.83	0.67	0.74
Yao et al. [26]	0.765	0.753	0.759
Shi et al. [21]	0.86	0.7	0.77
Xing et al. [23]	0.78	0.72	0.75
Xu et al. [24]	0.874	0.759	0.813
Ma et al. [14]	0.821	0.677	0.742
Zhou et al. [30]	0.835	0.671	0.744
Ours	0.813	0.715	0.761

Figure 5 presents some test results comparing the proposed method with EAST on ICDAR 2015. As shown, the bounding box generated by our method is more accurate than that of EAST. For some regions in which the text is densely distributed, our proposed method can detect more text than EAST.

Conclusion

In this paper, we proposed a scene text detection method through a convolutional neural network, which can detect multi-orientation text from natural scene images. Our proposed method uses an end-to-end pipeline, which makes the detection procedure simpler without lowering the accuracy. The proposed method achieved some competitive performance for multi-orientation text detection and long text lines in scene images, and the proposed method achieves an F-measure of 0.826 on ICDAR 2015 and 0.761 on MSRA-TD500, respectively. In additional, to improve the accuracy, the label and preprocessing of the testing processes can be strengthened.

Acknowledgments This work was supported by the Science and Technology Plan Projects of Shenzhen (No. JSGG20200807171601010, JSGG20191127151401743), the Graduate Education Reform Project of Shenzhen University (SZUGS2020JG11) and the grants from the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, China (H-ZG3K).

References

1. Ali A, Zhu Y, Zakarya M (2021) A data aggregation based approach to exploit dynamic spatio-temporal correlations for citywide crowd flows prediction in fog computing[J]. Multimed Tools Appl:1–33
2. Ch'ng CK, Chan CS (2017) Total-Text: A Comprehensive Dataset for Scene Text Detection and Recognition, pp 935–942. https://doi.org/10.1109/ICDAR.2017.157
3. Deng L, Gong Y, Lu X, Lin Y, Ma Z, Xie M (2019) STELA: a real-time scene text detector with learned anchor[J]. IEEE Access 7:153400–153407
4. Epshtein B, Ofek E, Wexler Y (2010) Detecting text in natural scenes with stroke width transform[C]. 2010 IEEE computer society conference on computer vision and pattern recognition. IEEE, pp 2963–2970
5. He P, Huang W, He T, Zhu Q, Qiao Y, Li X (2017) Single Shot Text Detector with Regional Attention. In Proc. of ICCV
6. Huang W, Qiao Y, Tang X (2014) Robust scene text detection with convolution neural network induced mser trees. In Proc. of ECCV
7. Huang G, Liu Z, Van Der Maaten L et al (2017) Densely connected convolutional networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA. IEEE, pp 4700–4708
8. Jaderberg M, Vedaldi A, Zisserman A (2014) Deep features for text spotting[C]//European conference on computer vision. Springer, Cham, pp 512–528
9. Jiang Y, Zhu X , Wang X, Yang S, Li W, Wang H, Fu P, Luo Z (2017) R2CNN: Rotational region CNN for orientation robust scene text detection. CoRR, vol. abs/1706.09579
10. Karatzas D, Gomez-Bigorda L, Nicolaou A et al (2015) ICDAR 2015 Competition on robust reading[C]// 2015 13th international conference on document analysis and recognition, IEEE, pp 1156–1160
11. Kim KH et al (2016) PVANET: Deep but lightweight neural networks for real-time object detection. arXiv preprint arXiv:1608.08021
12. Kingma DP, Ba J (2015) Adam: A Method for Stochastic Optimization[C]. International Conference for Learning Representations, San Diego, USA. 2015: CoRR abs/1412.6980
13. Liu W, Anguelov D, Erhan D et al (2016) Ssd: single shot multibox detector[C]// proceedings of European conference on computer vision, Springer, Cham, pp 21–37
14. Ma J, Shao W, Ye H, Wang L, Wang H, Zheng Y, Xue X (2018) Arbitrary-oriented scene text detection via rotation proposals[J]. IEEE Trans Multimed 20(11):3111–3122
15. Milletari F, Navab N, Ahmadi S A (2016) V-net: fully convolutional neural networks for volumetric medical image segmentation[C]. Proceedings of 2016 fourth international conference on 3D vision, Los Alamitos, CA, USA. IEEE Computer Society, pp 565–571
16. Nahari R, Putro S, Setiawan N, Alflta R (2020) Detecting Text in the Scene Text Image Using Fast Fourier Transform. Journal of Physics: Conference Series. 1569. 032070. https://doi.org/10.1088/1742-6596/1569/3/032070
17. Neumann L, Matas J (2010) A method for text localization and recognition in real-world images[C]. Asian conference on computer vision. Springer, Berlin, Heidelberg, pp 770–783
18. Ranjitha P, Rajashekar K, Shamjith (2020) A Review on Text Detection from Multi-Oriented Text Images in Different Approaches, pp 240–245. https://doi.org/10.1109/ICESC48915.2020.9156002

19. Ren S, He K, Girshick R et al (2015) Faster r-cnn: towards real-time object detection with region proposal networks[C]. Proceedings of the international conference on neural information processing systems, Istanbul, Turkey. pp 91–99
20. Rezatofighi H, Tsoi N, Gwak J Y et al (2019) Generalized intersection over union: a metric and a loss for bounding box regression[C]. Proceedings of conference on computer vision and pattern recognition, Long Beach, CA, USA. IEEE, pp 658–666
21. Shi B, Bai X, Belongie S (2017) Detecting oriented text in natural images by linking segments[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA. IEEE, 2550–2558
22. Tian Z, Huang W, He T et al (2016) Detecting text in natural image with connectionist text proposal network[C]//European conference on computer vision. Springer, Cham, pp 56–72
23. Xing D, Li Z, Chen X et al (2017) Arbitext: Arbitrary-oriented text detection in unconstrained scene[J]. arXiv preprint arXiv:1711.11249
24. Xu Y, Wang Y, Zhou W, Wang Y, Yang Z, Bai X (2019) TextField: learning a deep direction field for irregular scene text detection[J]. IEEE Trans Image Process 28(11):5566–5579
25. Yao C, Bai X, Liu W et al (2012) Detecting texts of arbitrary orientations in natural images[C]. Proceedings of the 2012 IEEE conference on computer vision and pattern recognition, Providence, RI, USA. IEEE, pp 1083–1090
26. Yao C, Bai X, Sang Net al (2016) Scene text detection via holistic, multi-channel prediction[J]. arXiv preprint arXiv:1606.09002
27. Ye Q, Doermann D (2014) Text detection and recognition in imagery: a survey[J]. IEEE Trans Pattern Anal Mach Intell 37(7):1480–1500
28. Yu J, Jiang Y, Wang Z et al (2016) Unitbox: an advanced object detection network[C]. Proceedings of the 24th ACM international conference on multimedia, New York, NY, USA. Association for Computing Machinery, pp 516–520
29. Zhang Z, Zhang C, Shen W et al (2016) Multi-oriented text detection with fully convolutional networks[C]. Proceedings of the 2016 IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA. IEEE, pp 4159–4167
30. Zhou X, Yao C, Wen H et al (2017) EAST: an efficient and accurate scene text detector[C]. Proceedings of the 2017 IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA. IEEE, pp 2642–2651
31. Zhu S, Zanibbi R (2016) A text detection system for natural scenes with convolutional feature learning and cascaded classification[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA. IEEE, pp 625–632
32. Zhu Y, Yao C, Bai X (2016) Scene text detection and recognition: recent advances and future trends[J]. Front Comput Sci 10(1):19–36

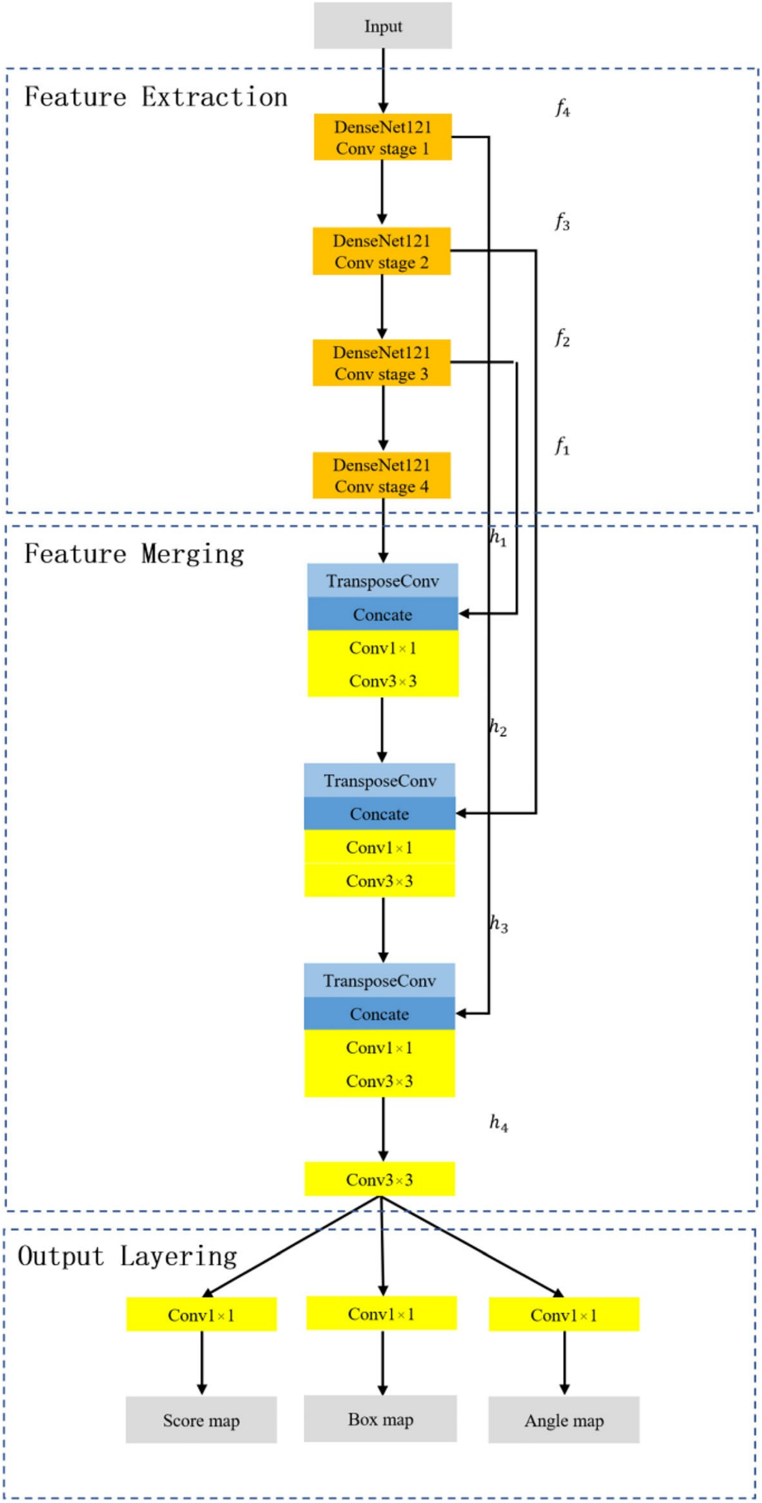
Fig. 1 Network structure of the proposed model

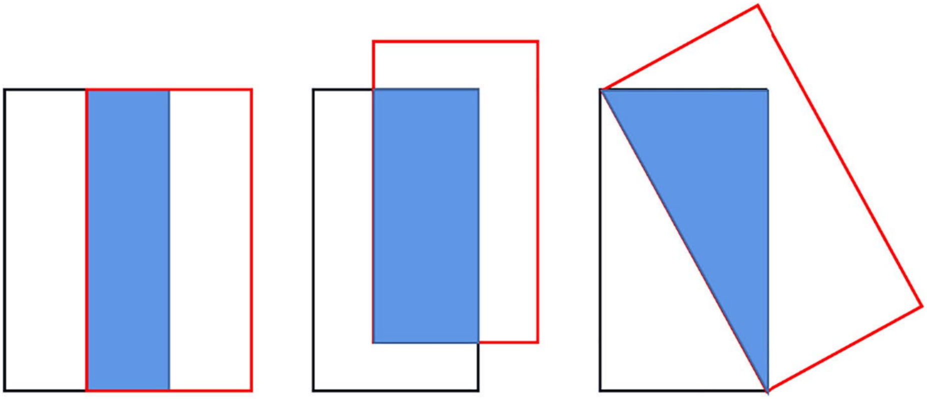
Fig. 2 Three cases when IoU is equal a Case 1 b Case 2 c Case 3

Fig. 3 Example for calculating GIoU (blue: ground-truth; red: predicted box; black dashed line: minimum cover bounding box)

Fig. 4 Different test results on ICDAR 2015 and MSRA-TD500 a Test results on ICDAR 2015 b Test results on MSRA-TD500

Fig. 5 Comparisons of results between EAST and our method (left: EAST; right: ours)

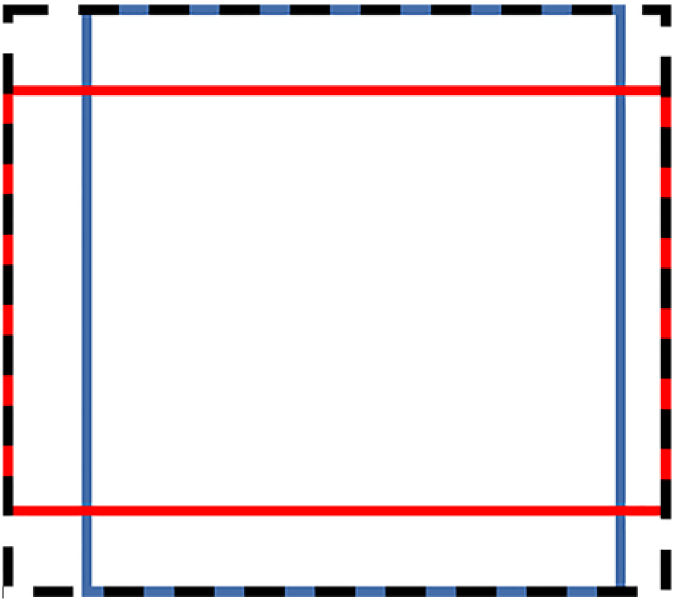

Input
Feature Extraction
DenseNet121 Conv stage 1
DenseNet121 Conv stage 2
DenseNet121 Conv stage 3
DenseNet121 Conv stage 4
f_4
f_3
f_2
f_1
Feature Merging
h_1
TransposeConv
Concate
Conv1×1
Conv3×3
h_2
TransposeConv
Concate
Conv1×1
Conv3×3
h_3
TransposeConv
Concate
Conv1×1
Conv3×3
h_4
Conv3×3
Output Layering
Conv1×1
Conv1×1
Conv1×1
Score map
Box map
Angle map



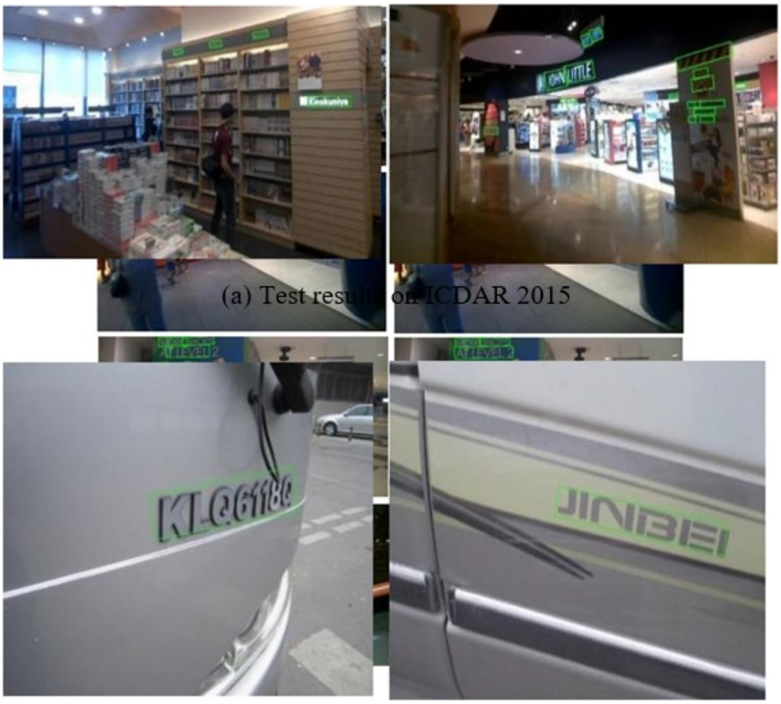
(a) Case 1

(b) Case 2

(c) Case 3



Method	**Precision**	**Recall**	**F-Measure**
Zhang et al. [27]	0.83	0.67	0.74
Yao et al. [28]	0.765	0.753	0.759
Shi et al. [23]	0.86	0.7	0.77
Xing et al. [29]	0.78	0.72	0.75
Xu et al. [24]	**0.874**	**0.759**	**0.813**
Ma et al. [26]	0.821	0.677	0.742
Zhou et al. [9]	0.835	0.671	0.744
Ours	0.813	0.715	0.761



(a) Test results on ICDAR 2015

(b) Test results on MSRA-TD500