Pattern Recognition xxx (xxxx) xxx

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Scale robust deep oriented-text detection network

Yuqiang Zheng[a], Yuan Xie[c,1], Yanyun Qu[a,*], Xiaodong Yang[a], Cuihua Li[a], Yan Zhang[a,b]

[a] Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, Fujian, China
[b] School of Big Data and Computer Science, Guizhou Normal University, China
[c] School of Computer Science and Technology, East China Normal University, Shanghai, China

ARTICLE INFO

Article history:
Received 6 May 2019
Revised 2 December 2019
Accepted 16 December 2019

Keywords:
Scene text detection
One-stage
Scale robust

ABSTRACT

Text detection is a prerequisite of text recognition, and multi-oriented text detection is a hot topic recently. The existing multi-oriented text detection methods fall short when facing two issues: 1) text scales change in a wide range, and 2) there exists the foreground-background class imbalance. In this paper, we propose a scale-robust deep multi-oriented text-detection model, which not only has the efficiency of the one-stage deep detection model, but also has the comparable accuracy of the two-stage deep text-detection model. We design the feature refining block to fuse multi-scale context features for the purpose of keeping text detection in a higher-resolution feature map. Moreover, in order to mitigate the foreground-background class imbalance, Focal Loss is adopted to up weight the hard-classified samples. Our method is implemented on four benchmark text datasets: ICDAR2013, ICDAR2015, COCO-Text and MSRA-TD500. The experimental results demonstrate that our method is superior to the existing one-stage deep text-detection models and comparable to the state-of-the-art text detection methods.

1. Introduction

Because text conveys high-level semantic information, extracting text information from natural scene images is increasingly demanded in numerous applications such as automatic driving, scene understanding, machine reading, and so on. Text detection is the prerequisite of text recognition and recently multi-oriented text detection becomes an active topic in the field of computer vision.

With the rising of deep learning [1], a breakthrough has been made in natural scene text detection. Most existing text detection methods based on deep learning have achieved prominent results which are much better than traditional text detection methods with a large margin [2,3]. According to Lyu et al. [4], they are divided into three categories: two-stage deep text detection, one-stage deep text detection, and deep text segmentation. Two-stage deep text detection methods evolve from Faster-RCNN [5] which firstly generate the proposal regions by Region Proposal Network (RPN) in the first stage and then classify the proposal regions using a convolutional neural network. One-stage deep text detection methods directly select proposal regions in feature maps rather than using RPN, which leads to high computational efficiency [6]. The representative methods are EAST [7] and TextBoxes++ [8] evolving from one-stage object detection methods such as SSD [9] and YOLO [10]. The third class of text detection methods implement deep semantic segmentation model for text detection, such as SegLink [11] and PixelLink [12]. It is recognized that the segmentation problem is more complex than the object detection problem. In Fig. 1, we give the comparison results of the three categories of text detection methods in terms of F-score and Frame Per Second (FPS). It is obvious that one-stage text-detection methods generally have an advantage in speed but have trailed the detection accuracy of two-stage text-detection methods. And deep text-segmentation methods are usually time-consuming with better text-detection accuracy.

Nevertheless, most existing text-detection methods are not robust when facing the two situations: 1) the text scales change in a large range. Few deep models can solve the scale-robustness in a single model. 2) The distribution of foreground-background classes is imbalanced, which lets down the detection accuracy of deep text-detection models. Furthermore, the one-stage deep text-detection methods have room to improve the text detection accuracy, though they have the potential to be fast and simple.

In this paper, we propose the Scale Robust Deep oriented-Text detection network (SR-Deeptext) which is robust to the change of text scales and mitigate the class imbalance. SR-Deeptext is a one-stage deep text-detection model. Due to the potential of EAST to be fast and superior in text detection performance, we choose EAST as our baseline. However, the downsampling

* Corresponding author.
E-mail addresses: zhengyuqiang@stu.xmu.edu.cn (Y. Zheng), yxie@sei.ecnu.edu.cn (Y. Xie), yyqu@xmu.edu.cn (Y. Qu), 449073626@qq.com (X. Yang), chli@xmu.edu.cn (C. Li), zy@gznu.edu.cn (Y. Zhang).
[1] Equal contribution.

https://doi.org/10.1016/j.patcog.2019.107180
0031-3203/© 2019 Elsevier Ltd. All rights reserved.

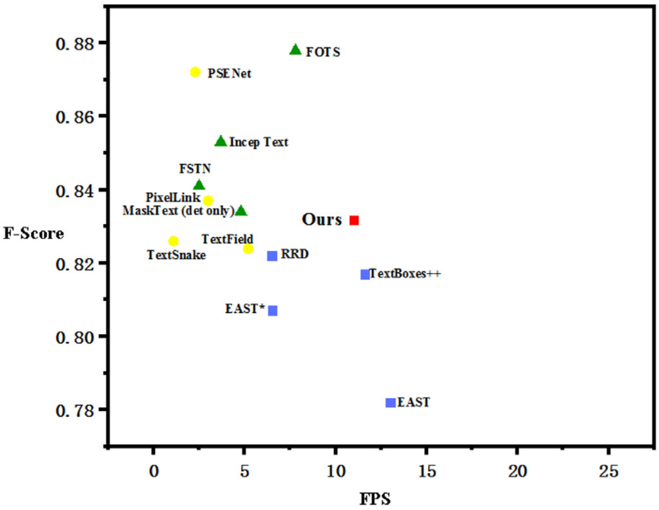


Fig. 1. Comparison of different methods on ICDAR2015. Yellow circles: segmentation-based methods; Green triangles: two-stage methods; Blue squares: one-stage methods; Red square: Our method. The one-stage methods are faster than most of two-stage methods and segmentation-based methods, but less accurate. Our method strikes a tradeoff between speed and accuracy. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

operation in EAST makes the feature maps smaller and the receptive field bigger, which causes that the feature maps of small size in the deeper layers go against the small object detection. Thus, we improve the feature fusion strategy to keep text detection in a high-resolution feature map. We choose the ResNet50 [13] as the backbone of our model. As it is observed that high resolution favors recognition in [12], we embed up-sampling layers in the network rather than multi-scale input, which avoids high computational complexity compared with multi-scale deep models. We design the refining block which includes the Residual Convolution Unit (RCU) and Chained Residual Pooling (CRP) to improve the prediction by using long-range residual connection. Moreover, in order to solve the foreground-background class imbalance problem, we use the Focal Loss [14] which focuses on the hard samples by down-weighting the well-classified proposal regions. As shown in Fig. 1, our method improves the detection accuracy compared with state-of-the-art one-stage text detection methods and achieves the better comprehensive criterion in detection accuracy and speed.

The contributions of our method are three-fold. 1) We propose the scale robust deep model for multi-oriented text detection, which not only has the potential of one-stage deep text-detection model to be fast but also has comparable accuracy to two-stage deep text-detection models. 2) The Focal Loss, which focuses on the hard samples by up-weighting the hard-classified samples, is employed in training the proposed deep model to avoid the foreground-background class imbalance. Our method avoids the foreground-background class imbalance. 3) Unlike the mainstream text-detection methods which use multi-scale input to deal with multi-scale problems, we design the feature refining block including the up-sampling layers, RCU and CRP to fuse multi-scale context features for keeping the text detection in the higher-resolution feature map, which leads to better detection accuracy.

The rest of this paper is organized in the following. In Section 2, related works on text detection are briefly reviewed. In Section 3, the proposed deep model is detailed. Experimental results are given and limitations are discussed in Section 4 and Section 5. Finally, we conclude this paper in Section 6.

2. Related work

In recent years, natural scene text detection has made great progress with the rapid development of deep learning. The research of text detection becomes deeper, and the study of text detection evolves from the horizontal text detection to the multi-oriented text detection [12,13] and further to the arbitrary-shape text detection [15,16]. As mentioned in Section 1, the existing deep text-detection methods can be divided into three categories: two-stage deep models, one-stage deep models and the deep segmentation models. In the following, we firstly make a brief introduction to the popular deep object-detection models and then introduce the three categories of text detection methods.

The existing deep object-detection deep models include two-stage deep models and one-stage deep models. Faster-RCNN [5] is a typical two-stage deep object-detection model, and in the first stage it obtains the candidate region of the target through RPN and then use a convolutional neural network to classify the candidate regions and make the position prediction. SSD [9] and YOLO [10] are famous one-stage deep frameworks for general object detection. They generate the proposal regions directly in feature maps rather than using RPN. One-stage object detection methods usually have high computational efficiency [6]. Most text detection methods evolve from the general object detection framework.

Two-stage text detection methods are draw from two-stage deep object-detection models which are charactered by RPN. IncepText [17] is proposed to use deformable PSROI for multi-oriented text detection, which makes the sampling field flexible and adaptable. Including the frequently-used loss functions: the classification loss and the position loss, it introduces the segmentation loss. However, the network speed is not satisfactory due to the introduction of deformable convolution. Fast oriented text spotting model (FOTs) [18] is a unified trainable deep model which simultaneously detects and recognizes the text and uses the traditional classification loss and regression loss. Fused text segmentation networks (FTSN) [19] is proposed to detect multi-oriented text, which uses a two-stage object detection module followed by a text instance segmentation module. The loss function is the summation of the RPN loss function, the classification loss function and the regression loss function. In addition, sliding line point regression (SLPR) [20] is a deep model for regressing the coordinates of the points on the edge of the text line, which is even effective at capturing text in arbitrary shapes. A new regression loss function is designed which is added to the traditional loss functions for text detection. IncepText and FOTs use multi-scale models to solve the problem of scale robustness. Moreover, all the mentioned methods use the multi-task learning to improve the accuracy of text detection.

Similarly, the one-stage text-detection deep models evolves from one-stage deep object-text models. EAST [7] merges the context information from multi-scale feature maps for dense per-pixel prediction and uses two schemes for the location prediction: rotated box (RBOX) and quadrangle (QUAD). RBOX can predict the text angle and distance from the pixel to the four boundaries of the minimum enclosing bounding box, while QUAD can directly predict the four corners of the text region. EAST becomes the baseline of many text detection methods. TextBoxes++ [8] also predicts two kinds of location information, but it inherits the idea of anchors in SSD to make the performance even much better. Moreover, TextBoxes++ improves the text detection by combining text recognition. RRD [21] extracts rotation-sensitive features and rotation-invariant features for position prediction and category determination, respectively. All the mentioned one-stage deep models use multi-scale-input for scale robustness and they neglect to discuss the foreground-background class imbalance.

Please cite this article as: Y. Zheng, Y. Xie and Y. Qu et al., Scale robust deep oriented-text detection network, Pattern Recognition, https://doi.org/10.1016/j.patcog.2019.107180

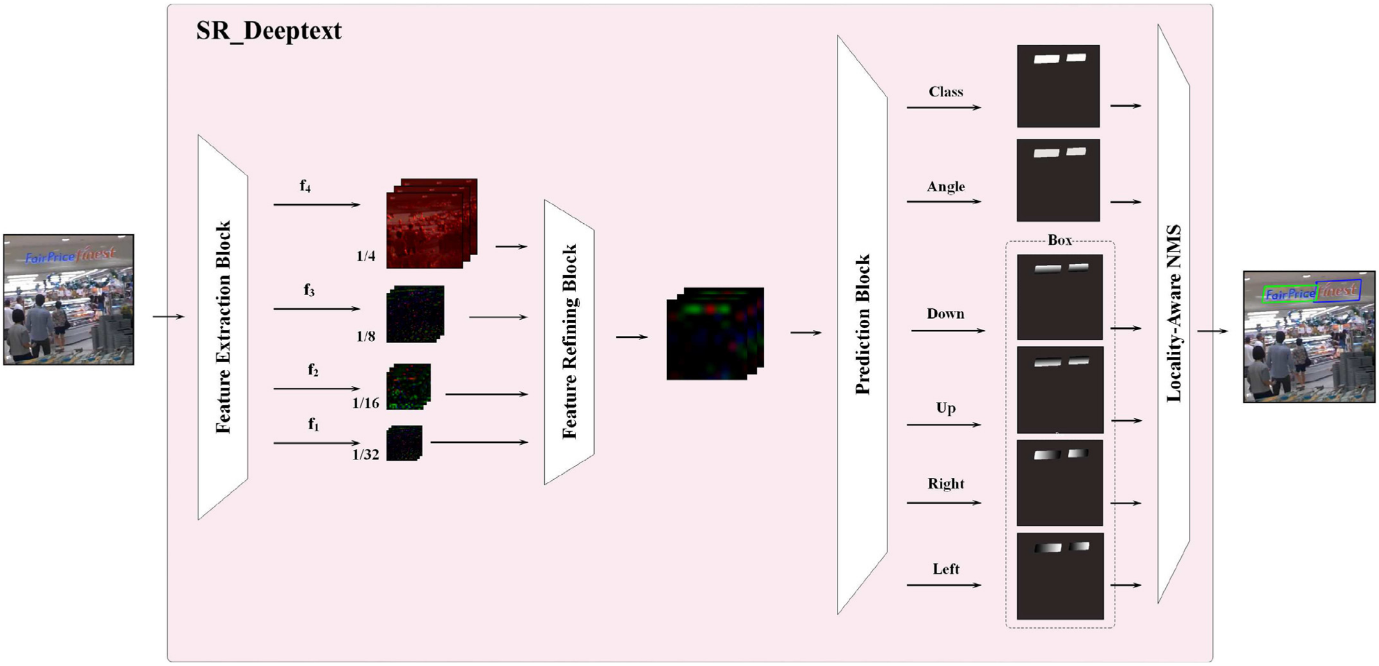


Fig. 2. The structure of our method. An image is fed into FFB to generate four feature maps (f_1, f_2, f_3, f_4). Then refined feature map is obtained by feature refining block with f_1, f_2, f_3 and f_4. The refined feature map is used to predict the confidence map, rotation angle map and geometry map. Finally, the location information of the text is obtained through locality-aware NMS.

Some methods regard the problem of text detection as the semantic segmentation of text regions. Seglink [11] gets the text box information by predicting text segmentation and the links between these segmentation. PixelLink [12] integrates multi-layer deep features to improve detection results. For the regular edge of text, Lyu et al. [4] proposed to predict the corner information of text while predicting the position-sensitive text segmentation. Recently, the arbitrary-shape text detection is newly raised and attracts more and more attention. MaskText [22] is a two-stage text detection method which unifies text mask segmentation. TextSnake [22] detects arbitrary shape text by predicting text centerlines and text regions. Similarly, PSENet [23] can achieve satisfactory results in detecting arbitrary shape text by fusing multi-level segmentation. TextField [24] learns the direction field for each text pixel which encodes the binary text mask and the direction information. TextField is actually a segmentation based method for arbitrary shape text detection.

To sum up, most of the two-stage and one-stage deep models for text detection deal with scale robustness by using multi-scale models, which leads to high computational cost. Moreover, the mentioned deep models for text-detection hardly discuss the foreground-background class imbalance. Thus, in this paper, we focus on solving the two problems.

3. Method

Though deep object-detection models have achieved prominent results, they cannot keep well detection performance if being implemented directly on natural scene text detection because text usually is a class of small objects which occupy not more than 10 percent of an image in area and change largely in appearance with diversity of fonts, rotations, scales and aspect ratios. It is recognized that the deep object-detection models are not adequately robust to scale variation [25]. Draw lessons from the designing methodology of EAST [7] which fuses the multi-scale context cues, we design a scale robust deep model for multi-oriented text detection. The architecture of SR-Deeptext is illustrated in Fig. 2 which contains three parts: the feature extraction block (FEB), the feature refining block (FRB), and the prediction block (PB). A query image is firstly fed into FEB, and then features are refined in FRB. Finally, the class score and the bounding box are output in PB. In the following, we introduce the implementation details of the three parts and the test scheme.

3.1. Feature extraction block

We employ ResNet50 [13] as the backbone which is proved experimentally to effectively improve the detection performance. The feature extraction block contains five convolutional blocks of ResNet50, i.e., conv1, conv2, conv3, conv4 and conv5. We remove the last average pooling layer and its following modules in ResNet50. Note that the features of small objects are lost with the increase of convolution layers due to downsampling. In order to retrieve the features of small objects, we fuse the context features in multiple convolutional blocks. The first retrieved feature map is gotten from the output of the pooling layers following *conv1*, and then the other multi-scale features for further fusion are obtained from the outputs of the last convolution layer in the first, second and fourth residual blocks, i.e., *conv2_3*, *conv3_4* and *conv5_3*. Thus, the four levels of feature maps are denoted as f_4, f_3, f_2, f_1 with the size of 1/4, 1/8, 1/16, 1/32 of the input image, respectively.

3.2. Feature refining block

The sequential downsampling in ResNet50 has two effects: 1) it widens the receptive field of the convolutions with the increase of the convolutional layers and captures more global and context information for better class prediction. 2) the downsampling operation makes the feature maps small and keeps the training fast and tractable. However, for the purpose of object detection, the feature maps of small size in the deeper layers go against the small object detection. In order to predict text proposal regions in a higher

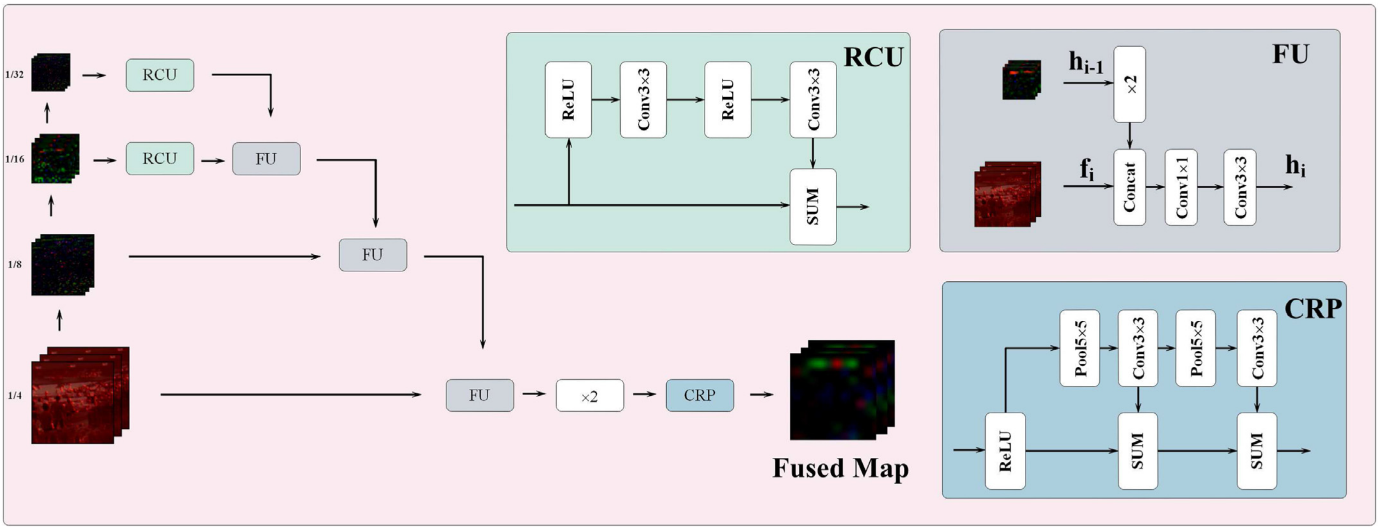


Fig. 3. The structure of Feature Refining Block. Two smaller feature maps (f_1, f_2) are preliminarily refined through the RCU. Then feature maps are fused from top to bottom. Finally, the fused features are refined again with CRP at high resolution.

resolution while keeping the large receptive field, we design the FRB to fuse the multi-scale features with a long-range connection between FEB and FRB. During the forward pass, the long-range connection passes the low-level features to encode the detail features, while in the training stage, the long-range connection is useful for the end-to-end training which allows direct gradient propagation.

Draw lessons from PixelLink [12], the high resolution images are beneficial to the detection. In other words, larger scale images usually achieve a better detection performance for one-stage scene text detection due to the larger size of the output [22]. We embed two special units in FRB: Residual Conv Unit (RCU) and Chained Residual Pooling (CRP) [26]. RCU is a simplified convolution unit from ResNet [13], which contains two Rectified Linear Units (ReLUs) and two convolutional layers without Batch Normalization (BN) layer. It fine-tunes the weights of the pre-trained ResNet50 for text detection in order to enhance the features by repeating activation and convolution operations, which is shown in Fig. 3.

CRP uses multi-scale windows to pool features, which is helpful to capture the background context. Fig. 3 gives the architecture of CRP which contains two units, each of which contains a maxpooing layer and a convolutional layer. The feature maps firstly pass through a ReLU layer, and then the following chain pooling is repeated two times, that is, the output is fed into two branches: the pooling layer followed by the convolution layer and the summation layer. CRP can make the proposed method reuse the information from the previous pooling operation.

The process of fusion in the Fusion Unit (FU) is formulated as,

$$h_i = \begin{cases} f_i & i = 1 \\ conv_{3\times3}(conv_{1\times1}([unsample(h_{i-1}); f_i])) & otherwise \end{cases} \quad (1)$$

where [· ; ·] represents the fusion operation which concatenates the feature maps along the third dimension of channel from small-scale feature maps to large-scale feature maps. All FUs have two input branches except the first stage. For the first feature block f_1, there is only the upsampling operation without the fusion operation. The extracted feature maps are passed through RCU and then up-sampled by 2 × factor. The output h_1 is fed into the second FU and is fused with the fine-tuned result of the second feature block f_2 by RCU. In the sequent two stages, only the fusion operation is done without the fine-tune operation by RCU. Finally, the fused feature block h_4 is up-sampled and pooled in CRP.

3.3. Prediction module

The prediction module is shown in Fig. 2. Actually, it contains several convolutional layers with 1 × 1 filter. The output feature maps of prediction block are projected into a single-channel score map F_s, a single-channel rotation angle map F_r and a multi-channel geometry map F_g. The rotation angle represents the orientation of the text bounding box. The geometry is represented by 4 channels of Axis-Aligned Bounding Box (AABB) [7] which denotes 4 distances (d_1, d_2, d_3, d_4) from the pixel location to the top, bottom, left, right boundaries of the minimum enclosing rectangle respectively. In the score map F_s, each pixel represents the confidence score s. In the rotation angle map F_r, each pixel represents the rotation angle θ of the box. During testing, a candidate pixel in the feature map whose score is larger than the threshold can generate a prediction box which depends on the corresponding coordinates x and y of the candidate pixel in the input image, and the predictors s, d_1, d_2, d_3, d_4 and θ in the output feature maps.

3.4. Loss functions

In this paper, we consider three losses: the loss for prediction score, the loss for rotation angle and the loss for geometry. The first loss measures the class prediction and the last two losses measures the regression of the text bounding box in the rotation angle and the four coordinates, and their corresponding loss functions are denoted by L_f, L_θ, and L_{AABB}, respectively.

It is recognized that the class imbalance will decrease the class-prediction performance of deep object-detection models. As for object detection with one-stage deep models, such as SSD [9], the foreground-background class imbalance greatly lowers the detection accuracy. Due to the large amount of candidate locations per image, which contains usually more than 10K candidates, training is not efficient. The negative candidate locations consume large computational resource. Moreover, the deep model is trained by the overwhelming easy negative samples favors to separate the easy negative samples from text samples while not distinguishing the hard-negative samples. Hard samples are frequently used to mitigate the class imbalance, which leads to a non-differential stage and more parameter-tuning.

Here, we employ Focal Loss for score map [14]. The well-classified samples are down weighted and the hard-classified samples are up weighted. Thus, Focal Loss will focus on training on a

Please cite this article as: Y. Zheng, Y. Xie and Y. Qu et al., Scale robust deep oriented-text detection network, Pattern Recognition, https://doi.org/10.1016/j.patcog.2019.107180

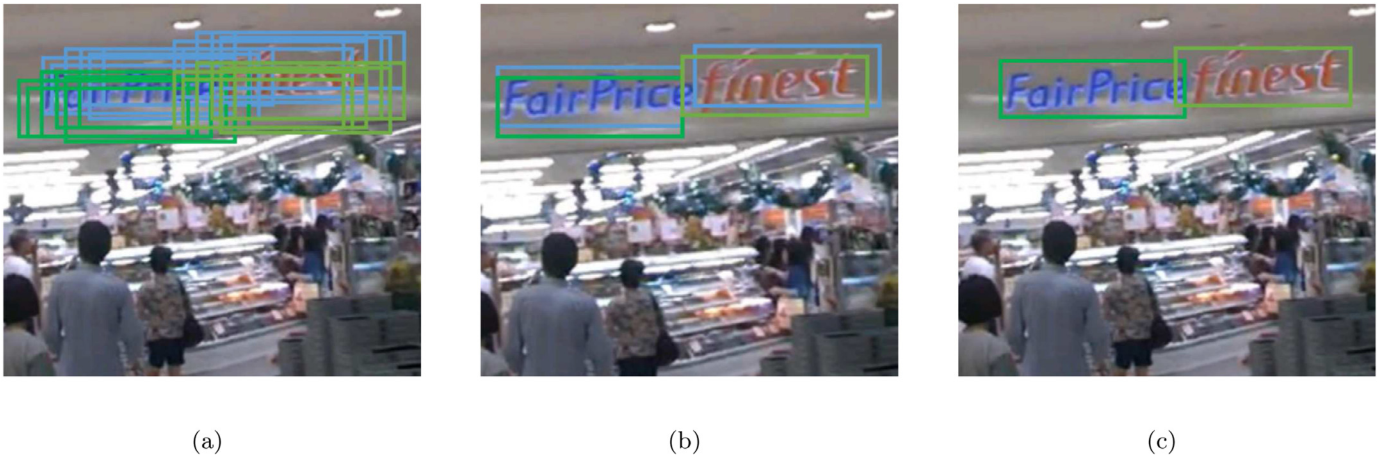


Fig. 4. The locality-aware NMS process. (a) The dense text boxes with different rows. (b) The dense text boxes are merged row by row. (c) The boxes are got through locality-aware NMS.

sparse set of hard samples and L_f is formulated as,

$$L_f = -(1-s_t)^{\gamma}\log(s_t) \tag{2}$$

$$s_t = \begin{cases} s & \hat{s} = 1 \\ 1-s & otherwise \end{cases} \tag{3}$$

where s is the text confidence score of the pixel, γ is set to 2, and $\hat{s}$ is the corresponding ground truth. In our experiments, we define the central area of a text ground truth region whose width and length are 70% of the minimum enclosing rectangle. The central area of the text ground truth is treated as the positive region and the remaining surrounding area without the positive region is treated as the negative region.

The loss for geometry L_{AABB} is introduced in [7] and is formulated as,

$$F_{AABB} = -\log\frac{R\cap\hat{R}}{R\cup\hat{R}} \tag{4}$$

where R represents the predicted region and $\hat{R}$ is its corresponding ground truth. R is defined as,

$$R = (d_1+d_3)*(d_2+d_4)$$

$$\hat{R} = (\hat{d_1}+\hat{d_3})*(\hat{d_2}+\hat{d_4})$$

$$R\cap\hat{R} = (\min(d_1,\hat{d_1})+\min(d_3,\hat{d_3}))*(\min(d_2,\hat{d_2})+\min(d_4,\hat{d_4}))$$

$$R\cup\hat{R} = R+\hat{R}-R\cap\hat{R}$$

where d_1, d_2, d_3 and d_4 represent distances from the pixel location to the top, bottom, left, right boundaries of the minimum enclosing rectangle respectively. And $\hat{d_1}, \hat{d_2}, \hat{d_3}$ and $\hat{d_4}$ are their corresponding ground truth. The orientation loss is introduced in [7] and calculated as,

$$L_{\theta} = 1-\cos(\theta-\hat{\theta}) \tag{5}$$

where θ is the prediction to the angle and $\hat{\theta}$ is its corresponding ground truth. The total loss function is formulated as,

$$L = L_f + L_{AABB} + \lambda_{\theta}L_{\theta}. \tag{6}$$

During training, λ_{θ} is set to 10.

3.5. Locality-aware NMS

During testing, dense text boxes are got from prediction module. And only correct text boxes are preserved via locality-aware NMS rather than the naive NMS. The latter is time-consuming with the computational complexity of $O(n^2)$, while the former runs only in $O(n)$. The process of locality-aware NMS contains three steps: 1) Traverse all boxes from adjacent pixels preferentially row by row under the assumption that the current box is highly correlated with its neighbor boxes. 2) Merge the adjacent boxes. Two adjacent boxes b and p need to be merged if the ratio of Intersection over Union (IoU) between b and p is greater than the threshold (the threshold of IoU is set to 0.1 in our experiments). The combination operation of b and p, which is formulated as,

$$s = s_b + s_p \tag{7}$$

$$x = \frac{(x_b*s_b + x_p*s_p)}{s} \tag{8}$$

where x_b, x_p and x represent coordinates of the boxes b, p and merging box, respectively. s_b, s_p and s represent the confidence score of the boxes b, p and the merging box. In Eq. (8), the coordinates of the merging box are the weight combination of the corresponding coordinates of the boxes b and p, and their confidence values of the boxes b and p are treated as their weight. In Eq. (7), the confidence score of the merging box is the sum of the two confidence scores corresponding to the two boxes. 3) Implement the naive NMS on the merged boxes. After the dense boxes are eliminated, the naive NMS is implemented to preserve the correct boxes. Fig. 4 shows the process of locality-aware NMS. In Fig. 4(a), the image contains the abundance of prediction boxes. If we use the naive NMS, the computational complexity is high. Fig. 4(b) shows the results of Step 2 after merging the adjacent boxes. Fig. 4(c) shows the final text boxes of Step 3 after implementing the naive NMS.

4. Experimental results

In order to evaluate the performance of our method, we implement SR-Deeptext on four public benchmark datasets: ICDAR2015 [27], MSRA-TD500 [28], ICDAR2013 [29] and COCO-Text [30]. We compare SR-Deeptext with the state-of-the-art text detection methods and then conduct ablation study to investigate the effects of RCU, CRP, and the Focal Loss function.

4.1. Benchmark datasets and measure

ICDAR2015 [27] is the dataset proposed on the Challenge 4 of ICDAR2015 Robust Reading Competition. The images are collected from photos of natural scenes in which the text foregrounds are

Please cite this article as: Y. Zheng, Y. Xie and Y. Qu et al., Scale robust deep oriented-text detection network, Pattern Recognition, https://doi.org/10.1016/j.patcog.2019.107180

Table 1
Comparison results on ICDAR2015. “Det”, “Seg”, “Reg” refer to “Detection”, “Segmentation ” and “Recognition” respectively. They indicate whether the method uses these tasks to train. “P”, “R”, “F” represent “Precision”, “Recall” and “F-score” respectively. Method with “*” means multi-scale testing. The best, second-best F-score are highlighted in red and blue, respectively.

Category	Method	Det	Seg	Reg	R(%)	P(%)	F(%)	FPS
Segmentation	PixelLink [12]		✓		82.0	85.5	83.7	3
	TextSnake [22]		✓		80.4	84.9	82.6	1.1
	PSENet [23]		✓		88.7	85.5	87.1	2.3
	TextField [24]		✓		80.5	84.3	82.4	5.2
	Corner [4]		✓		94.1	70.7	80.7	3.6
	Corner [4]*		✓		89.5	79.7	84.3	1
Tow-stage	Mask Textspotter [33]	✓	✓		81.2	85.8	83.4	4.8
	FOTS [18]	✓		✓	85.1	**91.0**	87.7	7.8
	FOTS RT [18]	✓		✓	79.8	85.9	82.7	24
	IncepText [17]	✓	✓		80.6	90.5	85.3	3.7
	SLPR [20]	✓			83.6	85.5	84.5	-
	FSTN [19]	✓	✓		80.0	88.6	84.1	2.5
One-stage	EAST [7]	✓			73.4	83.5	78.2	13
	EAST* [7]	✓			78.3	83.2	80.7	6.5
	TextBoxes+ [8]	✓			76.7	87.2	81.7	11.6
	TextBoxes+* [8]	✓			78.5	87.8	82.9	-
	RRD [21]	✓			79.0	85.6	82.2	6.5
	RRD* [21]	✓			80.0	88.0	83.8	-
	SR-Deeptext	✓			78.1	88.9	83.1	10.9

Table 2
Results on MSRA-TD500. “Det”, “Seg”, “Reg” refer to “Detection”, “Segmentation ”, “Recognition” respectively. “P”, “R”, “F” represent “Precision”, “Recall”, “F-score” respectively. The best, second-best F-score are highlighted in red and blue, respectively.

Category	Method	Det	Seg	Reg	R(%)	P(%)	F(%)	FPS
Two-stage	IncepText [17]	✓	✓		79.0	87.5	83.0	-
	FSTN [19]	✓	✓		77.1	87.6	82.0	-
Segmentation	Corner [4]		✓		76.2	87.6	81.5	5.7
	PixelLink [12]		✓		73.2	83.0	77.8	-
	SegLink [11]		✓		70.0	86.0	77.0	9
	TextField [24]		✓		75.9.3	87.4	81.3	-
One-stage	EAST [7]	✓			67.3	87.2	76.0	13.2
	SR-Deeptext	✓			74.4	84.6	79.2	8.6

multi-oriented, different types and cluttered backgrounds. It contains 1500 samples with 1000 training samples and 500 testing samples. A text region is annotated by the bounding box with the coordinates of the four corners. A sample is also labeled as an easy sample or a difficult sample. During testing, the score of the difficult sample can be excluded.

MSRA-TD500 [28] contains 500 images, including 300 training images and 200 testing images. The annotation information of a text region includes the coordinates of four positions and an orientation angle. And it also explains whether a sample is a difficult sample. Similar to EAST [7], we select 400 images of HUST-TR400 [31] to expand the training data due to the small number of dataset samples.

ICDAR2013 [29] contains 229 images for training, and 233 for testing. Different from ICDAR2015 and MSRA-TD500, the text instance is almost horizontal. The annotation information of text instance is given top-left coordinates, width, and height.

COCO-Text [30] is a large dataset which contains 43,686 images for training, 10,000 for validation and 10,000 for testing. Similar to ICDAR2013, it provides horizontal annotation information.

The benchmark measure for text detection relies on accounting to Precision (P), Recall (R), and F-score (F). They are given by:

$$P = \frac{TP}{TP + FP} \tag{9}$$

$$R = \frac{TP}{TP + FN} \tag{10}$$

$$F = 2 \times \frac{P \times R}{P + R} \tag{11}$$

where TP, FP and FN are the number of correct detection boxes, wrong detection boxes and missed detection boxes, respectively. A detected box *b* will be defined as the correct detection if the IoU is greater than 0.5 between *b* and the ground truth box. The performance of a text detection algorithm is generally measured by F-score and Frames Per Second (FPS).

4.2. Implementation details

SR-Deeptext is trained in an end-to-end way by using ADAM [32]. We randomly sample 512 × 512 crops from images and the batch size. The learning rate of ADAM is 0.0001, and the attenuation rate is 0.94 per 10,000 iterations. Training stops when the number of iterations reaches 150,000. During the test, the threshold of the text confidence score is set to 0.8. For ICDAR2015, we choose the pre-trained ResNet50 on ImageNet as the initial model. For other benchmark, we choose the pre-trained SR-Deeptext model in ICDAR2015 as the initial model. In the testing stage, we shrink the input image 0.5,0.8,0.8 time on MSRA-TD500, ICDAR2013 and COCO-Text, respectively. Our method is run on Tensorflow with GPU Geforce GTX 2080 Ti in Ubuntu 16.04 system.

4.3. Comparison with state-of-the-art methods

We compare our method with 19 text-detection methods including 13 state-of-the-art text-detection methods: EAST [7],

Table 3
Comparison results on ICDAR2013 and COCO-Text. “P”, “R”, “F” represent “Precision”, “Recall”, “F-score” respectively. The best, second-best F-score are highlighted in red and blue, respectively.

Method	ICDAR2013			COCO-Text		
	R(%)	P(%)	F(%)	R(%)	P(%)	F(%)
PixelLink [12]	83.6	86.4	84.5	-	-	-
SegLink [11]	83.0	87.7	85.3	-	-	-
Corner [4]	79.4	93.3	85.8	26.2	69.9	38.1
TextBoxes+ [8]	74.0	86.0	80.0	56.0	55.8	55.9
EAST [7]	-	-	-	32.4	50.4	39.4
SR-Deeptext	81.9	90.2	85.9	47.8	59.3	52.9

Table 4
Ablation study on different settings on ICDAR2015.

Method	Recall(%)	Precision(%)	F-score(%)
EAST [7]	73.5	83.6	78.2
+ResNet50	74.8	84.0	79.2
+Focal Loss	77.7	86.1	81.7
+RCU+CRP	76.2	88.5	81.9
SR-Deeptext	**78.1**	**88.9**	**83.1**

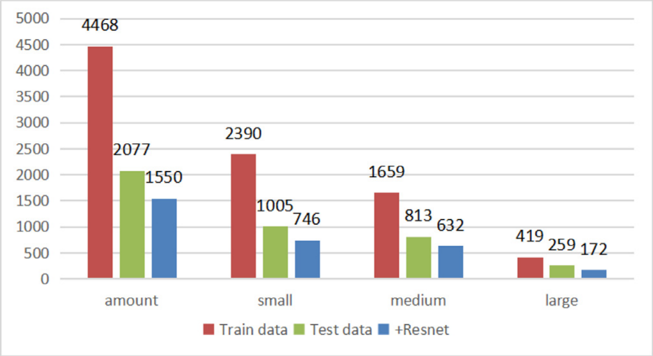


Fig. 5. The comparison of the amounts of text instances for training, testing and detecting in ResNet50 on ICDAR2015. The dataset consists mainly of small texts and medium texts, with only a small amount of large texts. Because of the class imbalance, the detection performance of the large texts is poor.

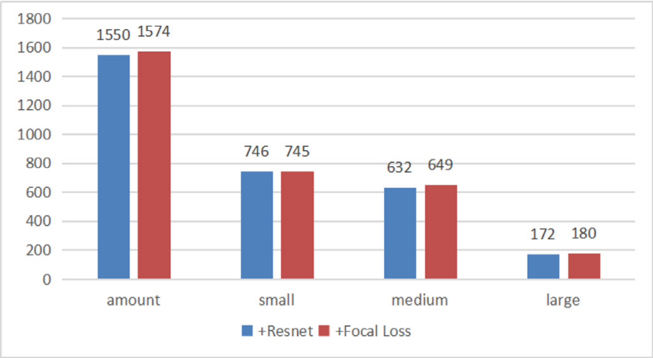


Fig. 6. The comparison of the amounts of detected text instances between +ResNet50 and +Focal Loss on ICDAR2015. The number of medium text and large text instances are increased. Training with Focal Loss can mitigate the sample imbalance problem.

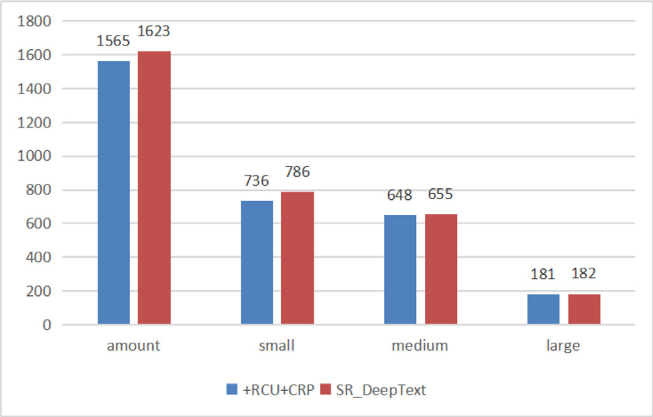


Fig. 7. The comparison of the amounts of detected text instances between +RCU+CRP and SR-Deeptext on ICDAR2015. Detection at the high resolution can be improved especially for small texts.

Table 5
Comparison of different loss functions with ResNet50 on ICDAR2015.

Method	Recall(%)	Precision(%)	F-score(%)
+ResNet50 (Original Loss)	74.8	84.0	79.2
OHEM [36]	76.3	84.4	80.2
Dice Loss [37]	77.3	84.6	80.8
Focal Loss	**77.7**	**86.1**	**81.7**

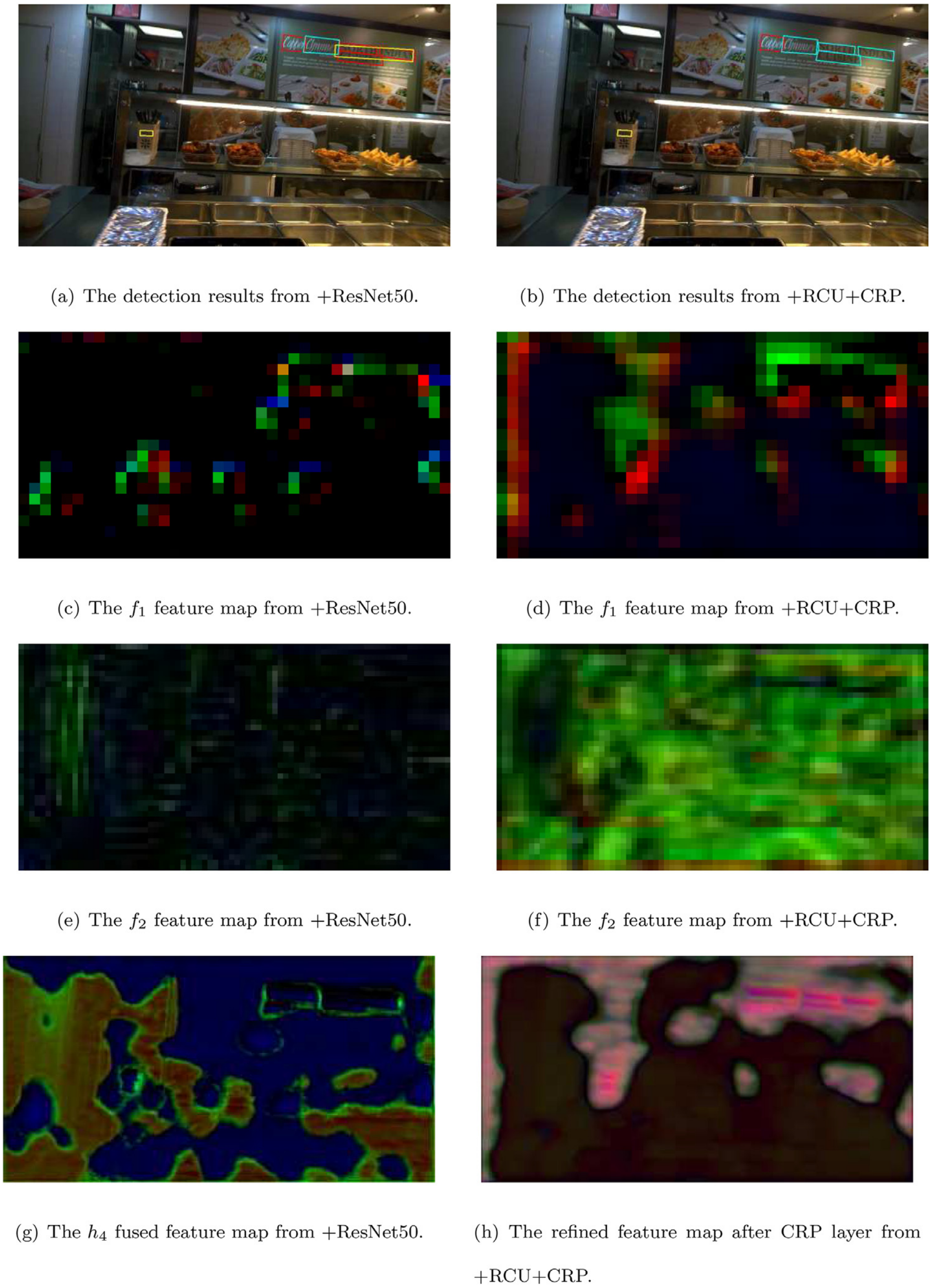
PixelLink [12], SegLink [11], TextBoxes++ [8], RRD [21], PSENet [23]. FOTS [18], Mask Textspotter [33], IncepText [17], SLPR [20], FSTN [19], TextField [24] and Corner [4], and together with their variants.

Table 1 shows the comparison results on ICDAR2015. All the results are the original published results. Because some competing methods use text recognition to improve the detection performance, for the fairness of comparison, we only compare the results in the detection part without regard of text recognition. In Table 1, we consider four important factors which greatly influence the detection performance: 1) does it use text segmentation? 2) does it benefit from text recognition? 3) what category is the method? 4) is it a multi-scale method?

Many text detection methods employ text segmentation scheme to improve the text detection performance, such as MaskText, PixelLink, TextSnake, PSENet, InceptText, FSTN, Corner, and its multi-scale scaling Corner*. Among them, PSENet achieves the best detection performance with the F-score of 87.1%. Recognition scheme is very helpful to text detection. FOTS achieves the highest text-detection F-score, which is 87.7%. As we mentioned in Section 1, most two-stage text detection methods are better than one-stage text detection methods in F-score but inferior to one-stage text detection methods in speed. Regarding the factor of scale, multi-scale text-detection methods achieve better detection performance than single-scale text-detection methods. The detection gain is over 1%, and the biggest gain is 3.6% which is obtained by Corner* compared with Corner. However, the multi-scale text-detection methods are time consuming compared with their corresponding single-scale text-detection methods, and their speeds are less than half of that of single-scale methods. EAST achieves the highest speed with 13 FPS.

Among the one-stage text methods such as EAST, TextBoxes++, RRD, our method achieves the highest F-score of 83.1% without regard of multi-scale methods. The speed of our method is 10.9 FPS. Our method ranks the third place in terms of the speed. Among Table 1, our method ranks the eighth place in F-score and ranks the fourth in speed. FOTS RT achieves the highest speed, which is a real-time variant of FOTS. FOTS RT is the variant of FOTS which is a two-stage method and is faster than the given one-stage methods, because FOTS RT uses ResNet34 as its backbone, while other two-stage methods given in Fig. 1 use ResNet50 as the backbone (FOTS, IncepText, MaskText), or ResNet101 as the backbone (FSTN). The real-time speed of FOTS RT attributes to the small backbone model. It can be seen that our method achieves the best

Please cite this article as: Y. Zheng, Y. Xie and Y. Qu et al., Scale robust deep oriented-text detection network, Pattern Recognition, https://doi.org/10.1016/j.patcog.2019.107180


(a) The detection results from +ResNet50.

(b) The detection results from +RCU+CRP.

(c) The f_1 feature map from +ResNet50.

(d) The f_1 feature map from +RCU+CRP.

(e) The f_2 feature map from +ResNet50.

(f) The f_2 feature map from +RCU+CRP.

(g) The h_4 fused feature map from +ResNet50.

(h) The refined feature map after CRP layer from +RCU+CRP.

Fig. 8. The comparison of feature maps between +ResNet50 and +RCU+CRP. For (a) and (b), Blue bounding boxes: correct detections; Yellow boxes: false detections; Red boxes: missed ground truths. The refined features become more differentiated between text regions and background. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

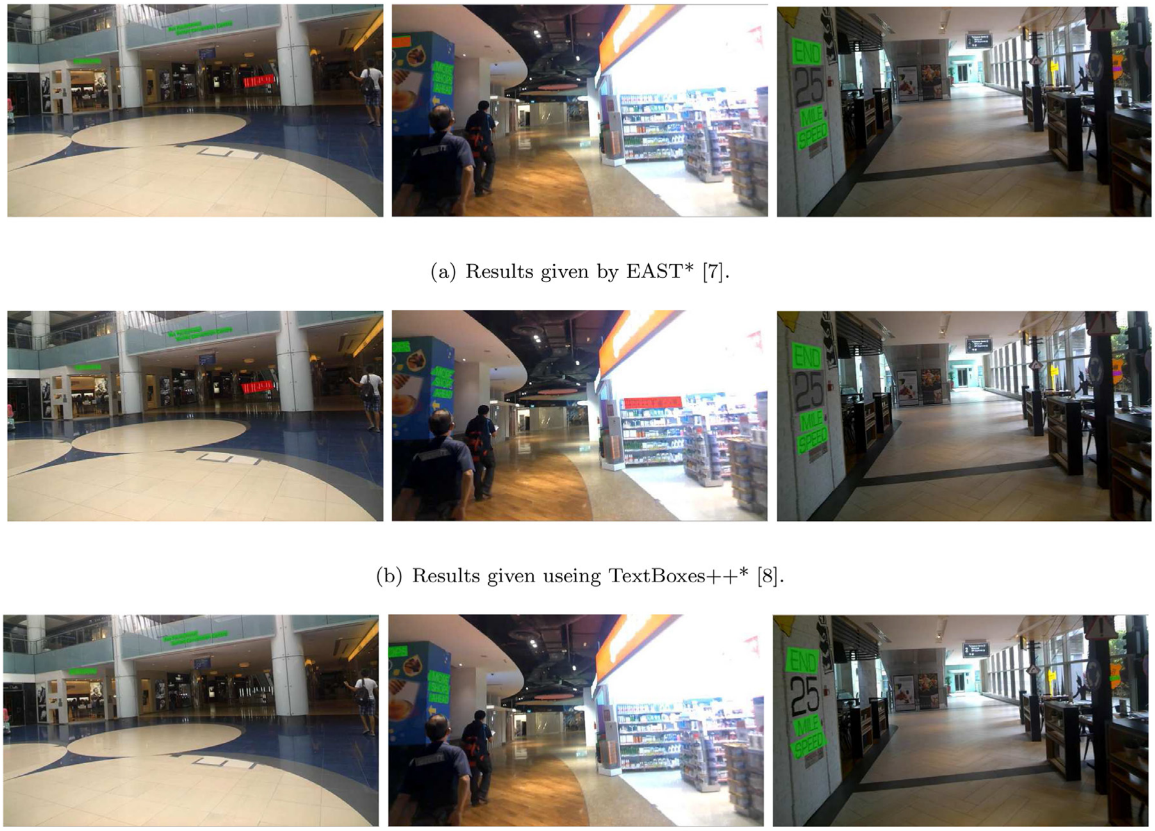
comprehensive performance in the competing state-of-the-art methods in ICDAR2015.

We also conduct experiments on MSRA-TD500 which is a harder dataset than ICDAR2015. The comparison results are shown in Table 2. Among the one-stage text detection methods, ours gains 3.2% in F-score compared with EAST. For the existing text detection methods on MSRA-TD500, IncepText achieves the best F-score and EAST achieves the highest speed. Our method ranks the fifth place in F-score and the third place in speed. Due to the large scale of text target and complicated text in MSRA-TD500, the existing text-detection methods obtain worse results on MSRA-TD500 than on ICDAR2015. Similar conclusion is made in MSRA-TD500, that is, the one-stage text-detection methods are superior in speed but not better in detection accuracy compared with the other two classes of text detection methods. Our method has gained 3.2% in F-score compare with EAST.

We have also evaluated our method on ICDAR2013 and COCO-Text which are popular horizontal text datasets. On ICDAR2013, our method achieves the best F-score of 85.9%. As shown in Table 3, our method ranks the second place in F-score on COCO-Text. It is worth noting that our method has gained 13.5% in F-score compare with EAST.

Fig. 9 and 10 show some examples of our method on ICDAR2015 and MSRA-TD500. In Fig. 9, we compare our method with EAST and TextBoxes++. It shows our method does not confuse the regular structures while the latter two methods wrongly regard the

Please cite this article as: Y. Zheng, Y. Xie and Y. Qu et al., Scale robust deep oriented-text detection network, Pattern Recognition, https://doi.org/10.1016/j.patcog.2019.107180



(a) Results given by EAST* [7].

(b) Results given useing TextBoxes++* [8].

(c) SR-Deeptext Results.

Fig. 9. Some comparisons of text detection results on ICDAR2015. Green bounding regions: correct detections; Red regions: false detections; Purple regions: missed ground truths. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

regular structure in the left image as the text region. TextBoxes++ also wrongly treats the books in the bookshelf in the middle image as the text region. In the right image, both EAST and TextBoxes++ miss a small text region. In Fig. 10, we show some example results of our method in ICDAR2015 and MSRA-TD500. Our method not only detects the English text but also the Chinese text with different fonts. Thus, our method is robust to the fonts and regular structure.

4.4. Ablation study

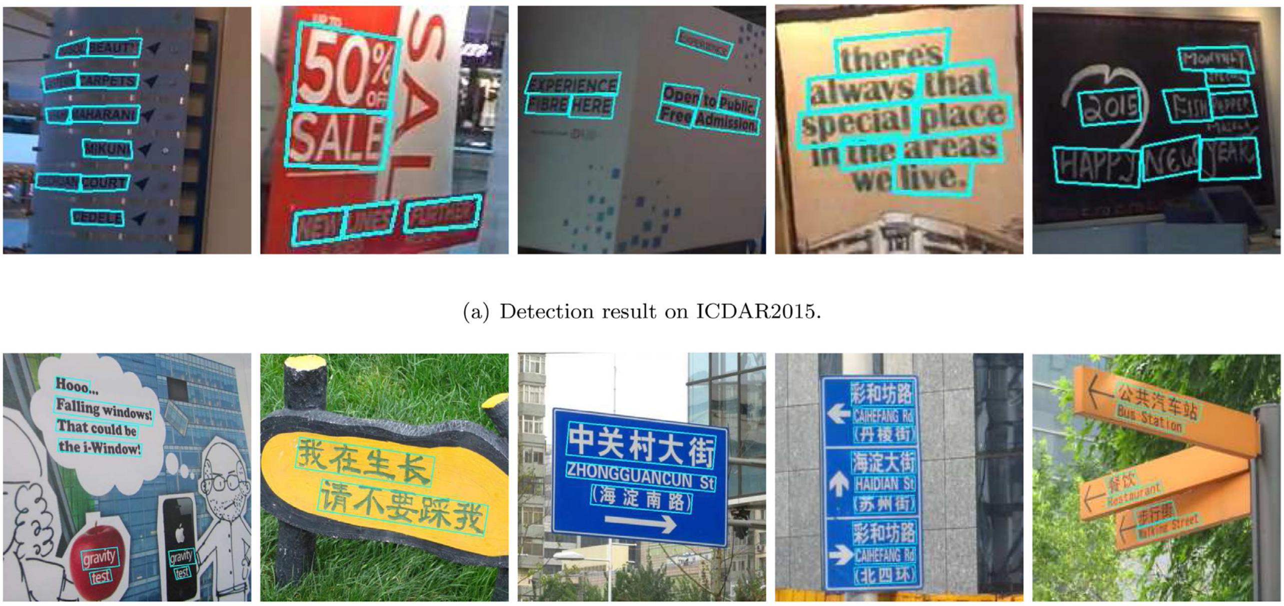
We conduct ablation study on ICDAR2015 to investigate the effect of five factors for multi-oriented text detection: the backbone network, FRB, the Focal Loss function and the scale. We construct three variants of our method: 1) +ResNet50: only ResNet50 is used and without regard of the feature fusion block and Focal Loss. 2) EAST: only the backbone of EAST is used with the common loss functions. 3) +Focal Loss: ResNet50 is used as the backbone with the Focal Loss function. 4) +RCU+CRP: it is combined with FRB but without magnifying the feature map with 2 × factor. The results of ablation study are shown in Table 4 and discuss the ablation study in the following.

We firstly show the imbalance of the text samples in scale. We make statistics about the text size to get the distribution of small-scale texts, medium-scale texts and large-scale texts according to the rule [34]. We empirically classify text instances into three groups according to their shorter side length: 1) small texts whose shorter side length are between 4 pixels and 24 pixels, 2) medium texts whose shorter side length are between 24 pixels and 48 pixels, and 3) large texts whose shorter side length are larger than 48 pixels. As shown in Fig. 5, most text targets are small texts and medium texts, while the large-scale text samples are relatively less which are only 9% of the total amount. Due to the text scale distribution imbalance, the variant +ResNet50 obtains higher detection performance in small and medium text targets with the detection recall of 74% and 77% while getting lower detection performance with the detection accuracy of 66% in large text targets.

The proportion of detection results of large text will be relatively little from the original method even the +ResNet50 is selected as the backbone network, due to the lesser number of large text from training samples.

The effect of backbone network. We compare two backbone networks: EAST and ResNet50. As shown in Table 4, ResNet50 is a little better than EAST in detection accuracy and recall. The gain of F-score is about 1%. It shows that the backbone network is important in feature extraction. ResNet50 extracts more distinctive features than EAST whose backbone is VGG16 [35].

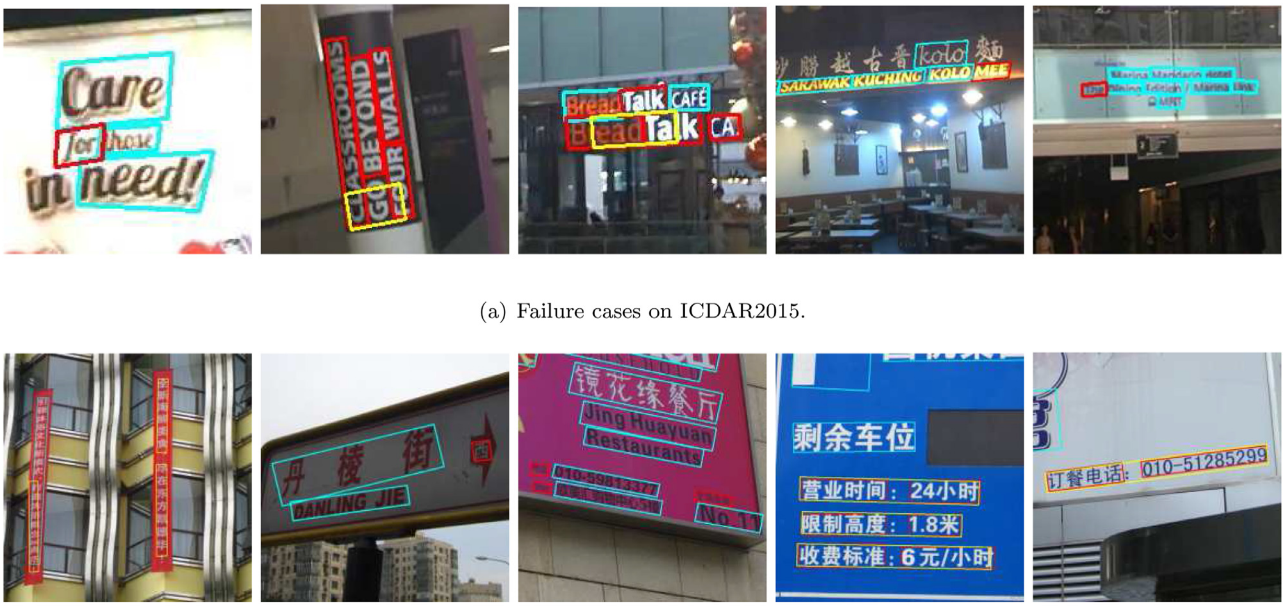
The effect of Focal Loss. We compare the two variants: +ResNet50 and +Focal Loss. As shown in Fig. 6, +Focal Loss correctly detects 17 more samples than +ResNet50 in medium text target and 8 more samples than +ResNet50 in large text targets. From Table 4, the gains of +Focal Loss are over 2% in the detection precision and F-score, respectively. We also compare four loss functions: the original loss of ResNet50, OHEM[32], Dice Loss [33], and Focal Loss. As shown in Table 5, our method with Focal Loss achieves the best F-score, and Dice Loss ranks the second, 0.9% lower than Focal Loss. OHEM ranks the third, and the original loss achieves the lowest F-score.


50% SALE
EXPERIENCE FIBRE HERE
Open to Public Free Admission
there's always that special place in the areas we live.
2015 HAPPY NEW YEAR
Hooo... Falling windows! That could be the i-Window!
gravity test
我在生长 请不要踩我
中关村大街 ZHONGGUANCUN St (海淀南路)
彩和坊路 CAIHEFANG Rd (丹棱街)
海淀大街 HAIDIAN St (苏州街)
彩和坊路 CAIHEFANG Rd (北四环)
公共汽车站 Bus Station
餐饮 Restaurant
步行街 Walking Street

(a) Detection result on ICDAR2015.

(b) Detection result on MSRA-TD500.

Fig. 10. Some examples of our method on ICDAR2015 and MSRA-TD500. Blue boxes: correct detections. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Failure cases on ICDAR2015.

(b) Failure cases on MSRA-TD500.

Fig. 11. Failure cases on ICDAR2015 and MSRA-TD500. Blue boxes: correct detections; Yellow boxes: false detections; Red boxes: missed ground truths. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The effect of RCU & CRP. We investigate the effect of RCU and CRP. +RCU+CRP is compared with the variant +Focal Loss. With the help of RCU and CRP, the detection accuracy of +RCU+CRP is significantly increased by 2.5% in Table 4, but the amount of recalled text targets slightly decreases.

We also compare the feature maps generated by +ResNet50 and +RCU+CRP in the Fig. 8. Fig. 8(a) and (b) show the detection results. The missed detection region is marked in red box, the error detection region is marked in the yellow box, and the correct detection is marked in the light blue box. +ResNet50 only correctly detects a text region while +RCU+CRP correctly detects almost all the text regions. Moreover, in Fig. 8(c)–(h), we give the visualization of the feature maps f_1, f_2, and h_4 of +ResNet50 and +RCU+CRP. It is observed that in the feature maps generated by ResNet50 the foreground colors are confused with those in the background. On the contrary, the colors in the feature map of +RCU+CRP are obviously different between the text area and background. Thus, +RCU+CRP enhances the discrimination between text from background.

The effect of upsampling. The difference between SR-Deeptext and +RCU+CRP is that the former contains the upsampling

Please cite this article as: Y. Zheng, Y. Xie and Y. Qu et al., Scale robust deep oriented-text detection network, Pattern Recognition, https://doi.org/10.1016/j.patcog.2019.107180

operation. As shown as Fig. 7, SR-Deeptext detects 50, 7 and 1 more text regions than +RCU+CRP in small, medium and large text detection, respectively. It demonstrates the effect of upsampling operation.

5. Limitations

The failure examples are shown in Fig. 11. It is observed that our method fails in short text containing only a few letters, e.g. 1–4. In addition, our method achieves poor detection performance on vertical text, because very small amount of vertical text samples exists in the training set. Furthermore, our approach cannot handle well the close-spaced texts, which may be caused by the ambiguous annotation.

6. Conclusions

Multi-oriented text detection in the wild is still a challenging task. Many deep models for text detection have achieved prominent results. One-stage text detection methods have the virtues of fast speed, but they are sensitive to text scale and the foreground-background class imbalance lets down the detection accuracy. In order to mitigate the two problems, we propose a scale robust deep model for multi-oriented text detection (SR-Deeptext). SR-Deeptext contains three parts: Feature Extraction Block, Feature Refining Block, and Prediction Block. ResNet50 is treated as Feature Extraction Block. RCU and CRP are embedded in Feature Refining Block together with upsampling operation. The long-range connection contained in FRB improves the discrimination between the foreground and the background. The Focal Loss is employed for training instead of the class-balanced cross entropy loss. We conduct extensive experiments to show our method is superior to the state-of-the-art methods in comprehensive performance of text detection. And among the one-stage text detection methods with a single scale, our method achieves the best detection performance. Moreover, the ablation study is conducted to show that the backbone network, FRB and the Focal Loss are all beneficial to the multi-oriented text detection.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant 61876161, Grant 61772524, Grant U1065252 and partly by the Beijing Municipal Natural Science Foundation under Grant 4182067, and partly by the Fundamental Research Funds for the Central Universities associated with Shanghai Key Laboratory of Trustworthy Computing.

References

[1] J. Gu, Z. Wang, J. Kuen, et al., Recent advances in convolutional neural networks, Pattern Recognit. 77 (1) (2018) 354–377.
[2] A. Zhu, R. Gao, S. Uchida, Could scene context be beneficial for scene text detection? Pattern Recognit. 58 (1) (2016) 204–215.
[3] Z. Zhong, L. Sun, Q. Huo, Improved localization accuracy by LocNet for Faster R-CNN based text detection in natural scene images, Pattern Recognit. 96 (1) (2019) 1–6.
[4] P. Lyu, Y. Cong, W. Wu, S. Yan, B. Xiang, Multi-oriented scene text detection via corner localization and region segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7553–7563.
[5] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach.Intell. 39 (6) (2017) 1137–1149.
[6] S. Zhang, L. Wen, X. Bian, Z. Lei, S.Z. Li, Single-shot refinement neural network for object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4203–4212.
[7] X. Zhou, Y. Cong, W. He, Y. Wang, S. Zhou, W. He, J. Liang, EAST: an efficient and accurate scene text detector, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2642–2651.
[8] M. Liao, B. Shi, B. Xiang, TextBoxes++: a single-shot oriented scene text detector, IEEE Trans. Image Process. 27 (8) (2018) 3676–3690.
[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, SSD: single shot multibox detector, in: European Conference on Computer Vision, 2016, pp. 21–37.
[10] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 779–788.
[11] B. Shi, B. Xiang, S. Belongie, Detecting oriented text in natural images by linking segments, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3482–3490.
[12] D. Dan, H. Liu, X. Li, C. Deng, PixelLink: detecting scene text via instance segmentation, in: The Thirty-Second AAAI Conference on Artificial Intelligence, 2018, pp. 6773–6780.
[13] K. He, X. Zhang, S. Ren, S. Jian, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[14] T. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Focal loss for dense object detection, IEEE Transactions on Pattern Analysis and MachineIntelligence. 1 (1) (2018). 1–1
[15] Y. Liu, L. Jin, S. Zhang, C. Luo, S. Zhang, Curved scene text detection via transverse and longitudinal sequence connection, Pattern Recognit. 90 (1) (2019) 337–345.
[16] J. Tang, Z. Yang, Y. Wang, Q. Zheng, Y. Xu, X. Bai, Detecting dense and arbitrary-shaped scene text by instance-aware component grouping, Pattern Recognit. 1 (1) (2019) 1–6.
[17] Q. Yang, M. Cheng, W. Zhou, Y. Chen, M. Qiu, W. Lin, W. Chu, IncepText: a new inception-text module with deformable PSROI pooling for multi-oriented scene text detection, in: Twenty-Seventh International Joint Conference on Artificial Intelligence, 2018, pp. 1071–1077.
[18] X. Liu, L. Ding, Y. Shi, D. Chen, Q. Yu, J. Yan, FOTS: fast oriented text spotting with a unified network, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5676–5685.
[19] Y. Dai, H. Zheng, Y. Gao, C. Kai, Fused text segmentation networks for multi--oriented scene text detection, in: International Conference on Pattern Recognition, 2018, pp. 3604–3609.
[20] Y. Zhu, J. Du, Sliding line point regression for shape robust scene text detection, in: International Conference on Pattern Recognition, 2018, pp. 3735–3740.
[21] M. Liao, Z. Zhen, B. Shi, G.S. Xia, B. Xiang, Rotation-sensitive regression for oriented scene text detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5909–5918.
[22] S. Long, J. Ruan, W. Zhang, H. Xin, W. Wu, Y. Cong, TextSnake: a flexible representation for detecting text of arbitrary shapes, in: European Conference on Computer Vision, 2018, pp. 19–35.
[23] L. Xiang, W. Wang, W. Hou, R.Z. Liu, L. Tong, Y. Jian, Shape robust text detection with progressive scale expansion network, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9336–9345.
[24] Y. Xu, Y. Wang, W. Zhou, Y. Wang, Z. Yang, X. Bai, TextField: learning a deep direction field for irregular scene text detection, IEEE Trans. Image Process. 18 (99) (2019) 5566–5579.
[25] B. Singh, L.S. Davis, An analysis of scale invariance in object detection snip, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3578–3587.
[26] G. Lin, F. Liu, A. Milan, C. Shen, I. Reid, RefineNet: Multi-path refinement networks for dense prediction, IEEE Transactions on Pattern Analysis and MachineIntelligence. 1 (1) (2019). 1–1
[27] D. Karatzas, S. Lu, F. Shafait, S. Uchida, et al., ICDAR 2015 competition on robust reading, in: International Conference on Document Analysis and Recognition, 2015, pp. 1156–1160.
[28] Z. Tu, M. Yi, W. Liu, B. Xiang, Y. Cong, Detecting texts of arbitrary orientations in natural images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1083–1090.
[29] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, et al., ICDAR 2013 robust reading competition, in: 2013 12th International Conference on Document Analysis and Recognition, IEEE, 2013, pp. 1484–1493.
[30] A. Veit, T. Matera, L. Neumann, J. Matas, S. Belongie, COCO-Text: dataset and benchmark for text detection and recognition in natural images, https://vision.cornell.edu/se3/coco-text-2/, 2016.
[31] Y. Cong, B. Xiang, L. Wenyu, A unified framework for multioriented text detection and recognition, IEEE Trans. Image Process. 23 (11) (2014) 4737–4749.
[32] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, in: International Conference on Learning Representations, 2015, pp. 1–15.
[33] M. Liao, P. Lyu, M. He, C. Yao, W. Wu, X. Bai, Mask TextSpotter: An end-to-end trainable neural network for spotting text with arbitrary shapes, IEEE Transactions on Pattern Analysis and MachineIntelligence. 1 (1) (2019). 1–1
[34] Z. Zhong, S. Lei, H. Qiang, An anchor-free region proposal network for faster R-CNN-based text detection approaches, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 315–327.
[35] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015, pp. 1–14.
[36] A. Shrivastava, A. Gupta, R. Girshick, Training region-based object detectors with online hard example mining, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 761–769.
[37] C.H. Sudre, W. Li, T. Vercauteren, S. Ourselin, M.J. Cardoso, Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations, in: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, Springer International Publishing, 2017, pp. 240–248.

Yuqiang Zheng received the B.S. degree in electronic information engineering from Fujian Normal University, China, in 2017. From 2017 to now, he is studying at Xiamen University for M.S. degree in computer science and technology. He is interested in the research of computer vision and image processing, object detection and image classification etc.

Yuan Xie (M'12) received the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences (CAS), in 2013. He is currently an full professor with the School of Computer Science and Software Engineering, East China Normal University. His research interests include image processing, computer vision, machine learning and pattern recognition. He has published around 35 papers in major international journals and conferences including the IJCV, IEEE TPAMI, TIP, TNNLS, TCYB, TCSVT, TGRS, TMM, and NIPS, CVPR, ECCV, etc. He also has served as a reviewer for more than 15 journals and conferences. Dr. Xie received the Hong Kong Scholar Award from the Society of Hong Kong Scholars and the China National Postdoctoral Council in 2014.

Yanyun Qu (M'12) received the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China, in 2006. She is currently a Professor in Computer Science Department, in School of Informatics, Xiamen University, Xiamen, China. She has published over 90 papers in major international journals and conferences, including the International Journal of Computer Vision, the IEEE Transactions on Image Processing, the IEEE Transactions on Cybernetics, and the IEEE Transactions on Geoscience and Remote Sensing, Neurocomputing, IEEE Conference on Computer Vision and Pattern Recognition, ACM International Conference on Multimedia, IEEE International Conference on Image Processing, and International Conference on Acoustics, Speech and Signal Processing. Her current research interests include image processing, computer vision, machine learning, and pattern recognition. Dr. Qu is a member of IEEE and ACM, and the Secretary of the Technical Committee of Hybrid Artificial Intelligence, Chinese Association of Automation.

Xiaodong Yang received the B.S. degree in law from Xiamen University, China, in 2015 and the M.S. degree in computer science and technology from Xiamen University, China, in 2018. His research interests include scene text detection and recognition.

Cuihua Li received the B.S. degree in computational mathematics from Shandong University, Jinan, China, in 1983, the M.S. degree in computational mathematics, and Ph.D. in automatic control theory and engineering from Xi'an Jiaotong University, Xi'an, China, in 1989 and 1999, respectively. He was an Associate Professor with the School of Science, Xi'an Jiaotong University before 1999. He is currently with the Department of Computer Science, Xiamen University, Xiamen, China. His current research interests include computer vision, video and image processing, and super-resolution image reconstruction algorithms. Prof. Li is a member of editorial boards of both Chinese Science Bulletin and Journal of Xiamen University natural science.

Yan Zhang received the B.S. degree in computer science and technology from Guizhou Normal University, China, in 2001 and the M.S. degree in software engineering from Guizhou University, China, in 2006. From 2015 to now, she is studying at Xiamen University for Ph.D. degree in Computer Science and Technology. She is interested in the research of computer vision and image processing, image classification and object detection etc.