Pattern Recognition 96 (2019) 106964

Contents lists available at ScienceDirect

Pattern Recognition

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

Rotated cascade R-CNN: A shape robust detector with coordinate regression

Yixing Zhu, Chixiang Ma, Jun Du*

National Engineering Laboratory for Speech and Language Information Processing University of Science and Technology of China, Hefei, Anhui, China

ARTICLE INFO

Article history:
Received 27 December 2018
Revised 30 June 2019
Accepted 10 July 2019
Available online 16 July 2019

Keywords:
Object detection
Text detection
Aerial images
Curved text
Rotated cascade R-CNN

ABSTRACT

General object detection task mainly takes axis-aligned bounding-boxes as the detection outputs. To address more challenging scenarios, such as curved text detection and multi-oriented object detection in aerial images, we propose a novel two-stage approach for shape robust object detection. In the first stage, a locally sliding line-based point regression (LocSLPR) approach is presented to estimate the outline of the object, which is denoted as the intersections of the sliding lines and the bounding-box of the object. To make full use of information, we only regress partial coordinates and calculate the remaining coordinates by the sliding rule. We find that regression can achieve higher precision with fewer parameters than the segmentation method. In the second stage, a rotated cascade region-based convolutional neural network (RCR-CNN) is used to gradually regress the target object, which can further improve the performance of our system. Experiments demonstrate that our method achieves state-of-the-art performance in several quadrangular object detection tasks. For example, our method yielded a score of 0.796 in the ICPR 2018 Contest on Robust Reading for Multi-Type Web Images, where we won first place for text detection tasks. The method also achieved 69.2% mAP on Task 1 of the ICPR 2018 Contest on Object Detection in Aerial Images, which was our best single model, where we also won first place. In addition, the method outperforms the previously published best record on the curved text dataset (CTW1500).

1. Introduction

Object detection and instance segmentation have attracted increasing attention in the computer vision community. The traditional object detection task mostly focuses on horizontal rectangular labeled objects, and instance segmentation focuses on arbitrary shape object segmentation. However, there are many tasks in which objects are labeled with a quadrangle or curved polygon, such as object detection in aerial images and text detection.

Recently, many researchers have adapted general object detection methods for object detection in aerial images and text detection. One type of method [1–4] directly regresses vertices of a quadrangular object, but this regression leads to ambiguity when we define the order of vertices. Another type of method is the instance segmentation method [5,6]. The instance segmentation based method can address the above mentioned ambiguity problem using mask labels but needs more parameters to increase the masks resolution, and mask prediction is not necessary for quadrangles and curved polygons.

For the shape of a quadrangle or a curved polygon, specific rules can be utilized compared with instance segmentation with an arbitrary shape. In this study, we propose a novel approach called locally sliding line-based point regression (LocSLPR) in which we use sliding lines to scan text line images of local proposal boxes and then regress the intersection points between the sliding line and ground-truth bounding box. There is no ambiguity in our method compared with directly regressing vertices. To make our approach more robust to the rotation problem, we further present the rotated cascade region-based convolutional neural network (RCR-CNN) in a two-stage manner. In the first stage, RoIAlign [7] is adopted, and the R-CNN network outputs LocSLPR's intersection points. Then, the rotated rectangle from the first stage is used to learn another rotated R-CNN, and rotated RoIAlign is adopted in the second stage. The proposed method can handle well objects whose label is quadrangle or curved polygon, which means our method is shape robust. The main contributions of our work are summarized as follows.

1. We present a novel LocSLPR method that can handle quadrangular/curved objects and well address the ambiguity problem of vertex order compared with direct regression. LocSLPR requires fewer parameters and achieves better results than segmentation-based methods.

* Corresponding author.
E-mail addresses: zyxsa@mail.ustc.edu.cn (Y. Zhu), ma1996@mail.ustc.edu.cn (C. Ma), jundu@ustc.edu.cn (J. Du).

https://doi.org/10.1016/j.patcog.2019.106964
0031-3203/

2. We present an RCR-CNN, which can gradually regress the object in a two-stage manner and significantly improves the performance of our system.
3. Our proposed method won first place in the ICPR 2018 Contest for Robust Reading for Multi-Type Web Images [8] with a score of 0.796 and was our best single model in the ICPR 2018 Contest on Object Detection in Aerial Images (ODAI) [9] with a 69.2% mean average precision (mAP), where we won first place. In addition, we also achieved the best results on the curved text detection dataset CTW1500 [10], demonstrating the effectiveness and flexibility of our method.

2. Related work

2.1. Object detection and instance segmentation

There are two main types of methods in object detection, namely, two-stage methods and one-stage methods. For the two-stage methods, Faster R-CNN [11] shares convolutional layers in the region proposal network (RPN) and R-CNN network; RPN proposes rough boxes in the first stage and then regresses again with R-CNN. R-FCN [12] presents a position-sensitive region of interest (RoI) pooling to learn the location information of objects. To address inaccurate localization problem, Wang et al. [13] presents a method named hierarchical objectness network for accurate localization. Hyperfusion-Net [14] tries to fuse reflective features which can integrate the global and local multi-scale feature maps. Tree-structured low-rank representation (TS-LRR) [15] presents a salient object detector which can improve the representation ability of the network for background, and distance the salient objects from the background. Ghadiri et al. [16] presents a novel method for detecting carried objects from a single video frame by incorporating multi-scales feature map. Cascade R-CNN [17] increases the number of R-CNN to gradually generate better boxes. However, these two-stage methods require a heavy computational load. Accordingly, a one-stage method is designed by removing the Fast R-CNN branch. YOLO [18] introduces a very fast framework that can process images in real time. SSD [19] generates multi-scale feature maps and detects the objects on the feature maps of different scales. In recent years, instance segmentation methods have also been widely applied to object detection. For example, Mask R-CNN [7] combines object detection with instance segmentation and presents RoIAlign to eliminate quantization error. The path aggregation network (PANet) [20], which won the COCO 2017 Challenge Instance Segmentation task, improves Mask R-CNN by bottom-up path augmentation, adaptive feature pooling and fully connected fusion. Since accurately annotated data is difficult to collect, weakly supervised learning is very important. Deep patch learning (DPL) [21] presents a novel method to learn patch features with only image-level annotations and proposal cluster learning (PCL) [22] also trains detector with only image-level annotations by generating proposal clusters for instance classifier refinement.

2.2. Text detection

First, general object detection methods can be applied to text detection tasks. In [23], a rotation region proposal network (RRPN) is proposed for multi-oriented scene text detection. R^2CNN [24] presents a multisize pool and regresses rotated rectangles in the R-CNN stage. The fused text segmentation network (FTSN) [5] improves Mask R-CNN for text detection. The CTW [10] regresses multiple points based on R-FCN for curved text detection and uses a recurrent neural network (RNN) [25] to learn the correlation between points. Liu and Jin [3] adopts SSD [19] to regress vertices and designs a rule to calculate the order of vertices. He et al. [26] presents a text attention mechanism (TAM) that roughly predicts text regions by an attention map. TextBoxes++ [27] adopts irregular 1×5 convolutional filters instead of 3×3 convolutional filters for long text lines and leverage recognition results to refine the detection results. SegLink [28] decomposes text into many parts, then predicts the probability of links, and finally merges them into one text line. Hu et al. [29] investigates detection of text lines on a character basis, which is different from word-level methods. WeText [30] presents a weakly supervised scene text detection method that is trained with unannotated or weakly annotated data. Based on pooling layer, Nguyen-Van et al. [31] presents a novel pooling based scene text proposal method for multi-orientation and multi-language scene text detection. Pastor [32] presents a novel text baseline detection method which is efficient and robust to nosily manuscripts. Text detection in mobile video is also a challenging task, Roy et al. [33] uses fractal property and optical flow for text detection in mobile video.

Second, text detection can also use segmentation-based methods. Liu and Jin [3] uses semantic segmentation to predict the salient map of text regions and the centroid of each character and then combines the two to restore text boxes. Wu and Natarajan [34] adds a border class to segmentation labels to separate nearby text-lines. PixelLink [35] generates an 8-direction margin to separate text lines. He et al. [36] generates segmentation maps of the text lines one by one with cascaded instance aware segmentation. Lyu et al. [37] combines position-sensitive segmentation with corner detection to calculate every quadrangle probability. Both Zhou et al. [2] and He et al. [1] combine segmentation and regression to generate a shrink score map and predict box border locations. He et al. [38] and Liu et al. [39] explore an end-to-end method for text detection and recognition based on EAST [2].

2.3. Object detection in aerial images

In the area of object detection in aerial images (ODAI), many researchers focus on transferring the powerful deep features from CNN to improve the performance of detectors for aerial images. Jiang et al. [40] and Chen et al. [41] use deep CNN features to detect small vehicles in satellite images. Similarly, Salberg [42] aims to detect seals in aerial remote sensing images with the help of off-the-shelf CNN feature representation. These methods simply replace traditional hand-crafted features with CNN features to acquire a richer representation to improve performance. Long et al. [43] divides ODAI into region proposal, classification, and accurate object localization. Hsieh et al. [44] attempts to use the correlation between objects based on the assumption that a predicted position where there are more objects can obtain a higher confidence to be predicted as the same object. Recently, one research direction focused on designing a unified deep detector for aerial images ([4] and [45]). Li et al. [46] presents a rotation-insensitive RPN and local-contextual feature fusion network for arbitrarily oriented instances, but its final result is also a horizontal bounding box (HBB). Cheng et al. [47] and Cheng et al. [48] focus on learning rotation-invariant CNNs for object detection. Although all these methods address the multi-oriented object detection, only Xia et al. [4] aims to detect oriented bounding boxes (OBBs) and presents faster-RCNN-OBB to directly regress vertices in R-CNN. With the popularity of machine learning, data-driven methods are widely used for the object detection task of aerial image datasets. To enlarge the data scale and diversity, Xia et al. [4] presents a large-scale dataset for object detection in aerial images (DOTA) including image samples with quadrangle labels from 15 categories.

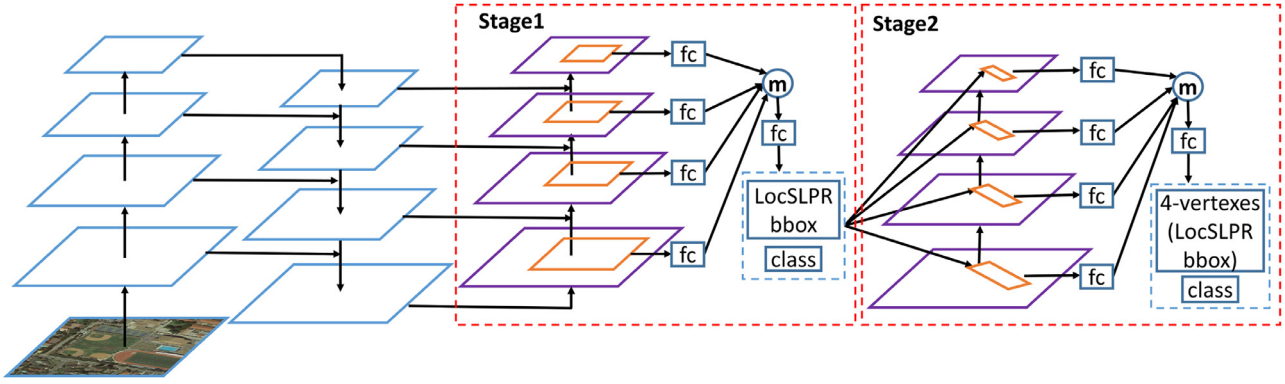


Fig. 1. Illustration of our proposed architecture.

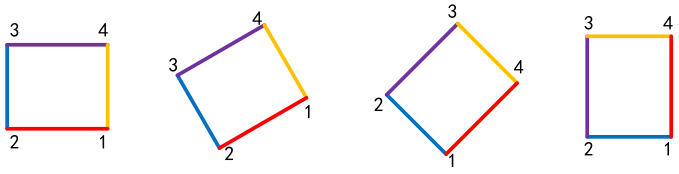


Fig. 2. Order ambiguity of vertices (note that the box's sides are painted different colors for better visualization).

3. Approach

3.1. Network architecture

The overview of the architecture is shown in Fig. 1. Inspired by PANet [20], we add both a top-down path and a bottom-up path to generate a feature pyramid with five feature maps, whose strides are 4,8,16,32,64, respectively. We use RPN to generate proposal boxes and assign anchors on feature maps with different scales. Specifically, the areas of 32×32, 64×64, 128×128, 256×256, and 512×512 pixels are set to 4-stride, 8-stride, 16-stride, 32-stride and 64-stride feature maps, respectively. The aspect ratios are 0.5, 1, and 2. These settings can refer to He et al. [7]. In the R-CNN stages, we use only 4-stride, 8-stride, 16-stride, and 32-stride feature maps. We extract four feature maps by RoIAlign algorithm on the feature maps with different scales, and then we add two fully connected layers and fuse the features of the four maps from the first fully connected layer by max pooling. We aim to regress arbitrary quadrangles, which is different from traditional object detection tasks. To avoid the ambiguity of vertex order, we use LocSLPR to generate the outline of the objects. To generate a more accurate box, we employ RCR-CNN with two stages. In the first R-CNN, the inputs are horizonal rectangles and ouputs are rotated rectangles, while in the second R-CNN, the inputs are rotated rectangles calculated with the outputs from the first stage. As the ambiguity problem of vertex order is well solved by LocSLPR in the first R-CNN, we directly regress four quadrangular vertices in the second R-CNN for quadrangular objects and still use LocSLPR for curved texts. In the following subsections, two main contributions, namely, LocSLPR and RCR-CNN, will be explained.

3.2. LocSLPR

A quadrangle is made up of four vertices. Although we can directly regress these vertices, we need to formulate a rule to determine the order of the four vertices. As shown in Fig. 2, if we define the vertex that is closest to 45 degrees as the first vertex, ambiguity will appear near 45 degrees. This order ambiguity makes it difficult for the network to learn, which is the motivation for the proposed LocSLPR.

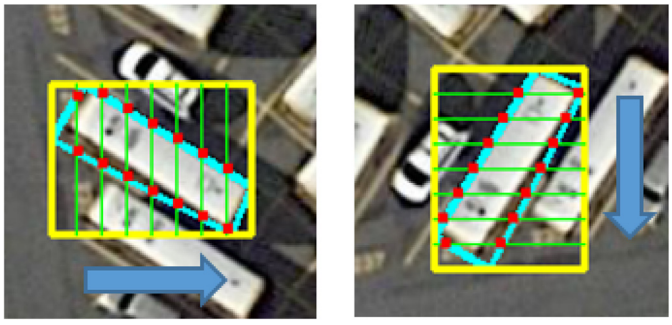


Fig. 3. The intersection generation process of LocSLPR using horizontal sliding (left) and vertical sliding (right); horizontal/vertical sliding refers to the direction of the line scan but not the angle of the line. If a box's width is larger than its height, horizontal sliding lines are used; otherwise, vertical sliding lines are used.

Different from our previous work [49], LocSLPR slides lines along each proposal box rather than the target object box and then regresses the intersections of the sliding lines and the object border. We show the intersection generation process in Fig. 3. Then, the coordinates of the intersection points are calculated. For horizontal sliding with uniformly spaced lines, we can easily obtain the x-coordinate of the intersection points from the position of the proposal box and only need to regress the y-coordinate of these points. For vertical sliding with uniformly spaced lines, we can easily obtain the y-coordinate of the intersection points from the position of the proposal box and only need to regress the x-coordinate of these points. Thus, we not only reduce the system parameters but also restrain the regressing points, which will generate a more regular polygon [49]. We find that the intersection points along the long side of the proposal box can better represent the outline of an object. Accordingly, we use only these points to restore the object.

We define the loss function of the LocSLPR regression task as:

$$L_{\text{LocSLPR}} = \frac{1}{4n}\left[I\left(\frac{w_{\text{p}}}{h_{\text{p}}} < \frac{1}{r}\right) \sum_{j=1}^{2n} \text{smooth}_{L_1}(\bar{x}_{v_j}, \bar{x}^*_{v_j}) + I\left(\frac{w_{\text{p}}}{h_{\text{p}}} > r\right) \sum_{i=1}^{2n} \text{smooth}_{L_1}(\bar{y}_{h_i}, \bar{y}^*_{h_i}) \right] \quad (1)$$

where n is the number of sliding lines, w_{p} is the width of the proposal box, and h_{p} is the height of the proposal box. $I(\cdot)$ is the indicator function. As shown in Fig. 3, we can well represent an object by a sliding line on the long side. Therefore, we add weight r to Eq. (1); r is a threshold for the aspect ratio of the proposal box, which is set to 0.8. Thus, the network only regresses the intersec-

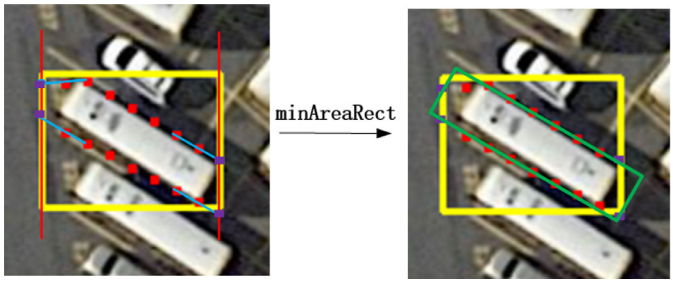


Fig. 4. Calculating the rotated rectangle from LocSLPR's points.

tion points on the longer side. $\bar{x}_{v_j}$ is the x-coordinate of the intersection point v_j of the vertical sliding lines and the object outline, while $\bar{y}_{h_i}$ is the y-coordinate of the intersection point h_i between the horizontal sliding lines and the object outline. $\bar{x}^*_{v_j}$ and $\bar{y}^*_{h_i}$ are the corresponding versions predicted by the network. smooth_{L_1} denotes the Smooth L_1 loss in [50]:

$$\text{smooth}_{L_1}(z, z^*) = \begin{cases} 0.5(z - z^*)^2 & \text{if } |z - z^*| < 1 \\ |z - z^*| - 0.5 & \text{otherwise} \end{cases} \tag{2}$$

Please note that all the above coordinates are normalized on the proposal box with the center coordinates (x_p, y_p) as:

$$\begin{aligned} \bar{x}_{v_j} &= (x_{v_j} - x_\text{p})/w_\text{p}, \quad \bar{x}^*_{v_j} = (x^*_{v_j} - x_\text{p})/w_\text{p} \\ \bar{y}_{h_i} &= (y_{h_i} - y_\text{p})/h_\text{p}, \quad \bar{y}^*_{h_i} = (y^*_{h_i} - y_\text{p})/h_\text{p} \end{aligned} \tag{3}$$

As shown in Fig. 4, we calculate the rotated rectangle from LocSLPR's points. First, we connect the two points closest to the horizontal rectangle's short side and then extend and calculate their intersection and the short side. Next, we calculate the minimum rotated rectangle that includes these points. When the number of sliding lines is small, it is possible to output a longer rectangle, but the angle of the rotated rectangle is always accurate. Due to the second regression in our cascade R-CNN, we find that this small number does not have a negative impact, so we set the number of sliding lines in the first stage to 7 to reduce the amount of calculations. If there is no next stage, the number of sliding lines is set to 28.

3.3. Rotated cascade R-CNN

In recent years, many methods [23,38,39,46] have explored rotated proposal boxes and RoIRotate. Inspired by cascade R-CNN [17], we present RCR-CNN. We adopt R-CNN twice, the inputs are horizontal rectangles in the first stage and rotated rectangles in the second stage. Accordingly, we calculate IoU for ground-truth matching on horizontal rectangles in the first stage and on rotated rectangles in the second stage. This process is shown in Figs. 5 and 6.

Rotated RoIAlign (RRoIAlign) is inspired by RoIAlign [7]; RRoIAlign can process a rotated rectangle box that is more suitable for our task. As illustrated in Fig. 7, RoIAlign adopts bilinear interpolation to compute the input featuresâ values at four sampled locations in each RoI bin and then calculates the results by using each bin's average. Our RRoIAlign follows the rule of RoIAlign with an additional angle variable. RoIPool and RRPN [23] use quantizations leading to the offset. These misalignments might have a negative effect on the regression, especially for small objects. Therefore, we adopt bilinear interpolation to calculate the values of these points.

To implement the RRoIAlign for region proposal, we change the label computation method. As shown in Fig. 8, we build the new coordinate system by setting the long side of the proposal box to the x-axis and the short side to the y-axis. Suppose that (x', y') is the original coordinate system that is the original label and that (x, y) is the transformed coordinate system. To transform between two coordinate systems, we first translate the original coordinates by $(-x_\text{p}, -y_\text{p})$, then rotate the coordinates by θ degrees, and finally translate the coordinates back by (x_p, y_p). The above operations can be represented by an affine matrix:

$$\begin{aligned} \boldsymbol{M} &= \begin{bmatrix} 1 & 0 & x_\text{p} \\ 0 & 1 & y_\text{p} \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -x_\text{p} \\ 0 & 1 & -y_\text{p} \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta & \sin\theta & (1-\cos\theta)x_\text{p} - y_\text{p} * \sin\theta \\ -\sin\theta & \cos\theta & (1-\cos\theta)y_\text{p} + x_\text{p} * \sin\theta \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4}$$

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \boldsymbol{M} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \tag{5}$$

Based on Eqs. (4) and (5), ground truth (x, y) on the rotated coordinate system can be calculated by ground truth (x', y') on the original coordinate system, we use the ground truth (x, y) on the rotated coordinate system to calculate the regression target in the second stage. By using RRoIAlign, we need to further adjust the coordinates only in the second stage. Then, the ambiguity problem is also accordingly well addressed. Since generating quadrangles from contours is an iterative process, we directly regress four vertices in the second stage to accelerate the quadrangular task; the order of the four vertices is determined by using the following rule: we first sort these vertices clockwise and then calculate the mean values of these vertices' coordinates as the center point. Accordingly, four vectors are formed by linking from the center point to the four vertices. Finally, the angles of these vectors can be computed, and the vertex that is closest to 45° is selected as the starting point. However, for the curved text task, we still use LocSLPR in the second stage. We define the regression loss function of the four vertices as:

$$L_{4\text{P}} = \sum_{i=1}^{4} \left[\text{smooth}_{L_1}(\bar{x}_i, \bar{x}^*_i) + \text{smooth}_{L_1}(\bar{y}_i, \bar{y}^*_i) \right] \tag{6}$$

where $\bar{x}_i$ and $\bar{y}_i$ are the x-coordinate and y-coordinate of the ith vertex, $\bar{x}^*_i$ and $\bar{y}^*_i$ are the corresponding values predicted by the network, and smooth_{L_1} is the smooth L_1 loss defined in Eq. (2). Similar to Eq. (3), $\bar{x}_i, \bar{x}^*_i, \bar{y}_i, and \bar{y}^*_i$ are normalized coordinates:

$$\begin{aligned} \bar{x}_i &= (x_i - x_\text{p})/w_\text{p}, \quad \bar{x}^*_i = (x^*_i - x_\text{p})/w_\text{p} \\ \bar{y}_i &= (y_i - y_\text{p})/h_\text{p}, \quad \bar{y}^*_i = (y^*_i - y_\text{p})/h_\text{p} \end{aligned} \tag{7}$$

For the classification task, the loss function is defined as:

$$L_\text{classes} = \sum L_{cls}(y_\mathbf{c}, y^*_\mathbf{c}) \tag{8}$$

where L_{cls} is a cross-entropy loss function, $y^*_\mathbf{c}$ is the ground truth of classification, and $y_\mathbf{c}$ is the prediction score.

4. Experiments

4.1. Object detection in aerial images

4.1.1. DOTA

DOTA is a large-scale dataset for object detection in arial images [4], which contains 2806 aerial images from different sensors and platforms. The size of these images ranges from 800×800 to 4000×4000 pixels, and there are large scales and angle spans between objects. Fifteen common object categories, namely, plane, ship, storage tank (ST), baseball diamond (BD), tennis court (TC), basketball court (BC), ground track field (GTF), harbor, bridge, large vehicle (LV), small vehicle (SV), helicopter (HC), roundabout (RA),

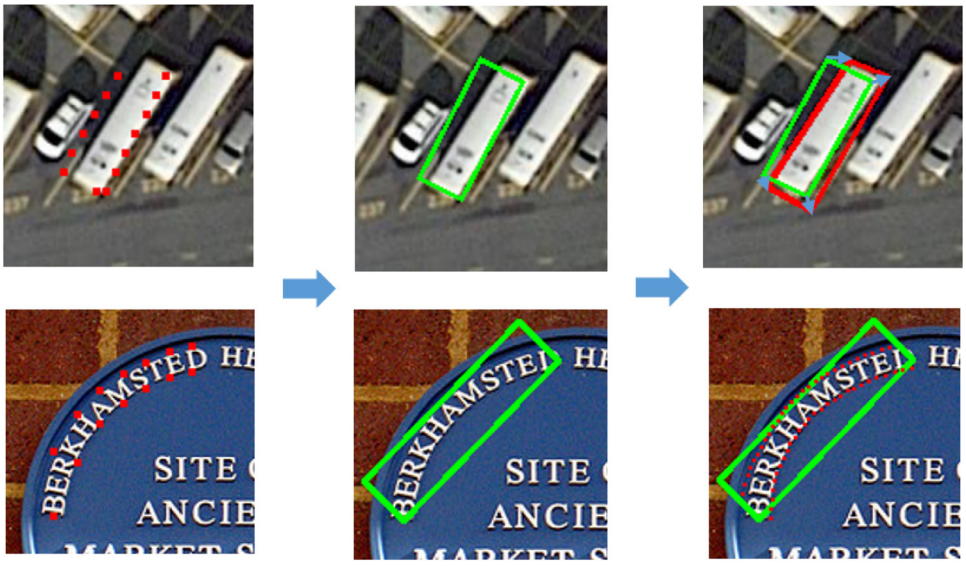


Fig. 5. The process of RCR-CNN. From left to right: The LocSLPR points (red) along the long side in the first stage, the rotated rectangle (green) generated by the LocSLPR points, and the regression in the second stage (note that only part of the red points are shown for better visualization). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

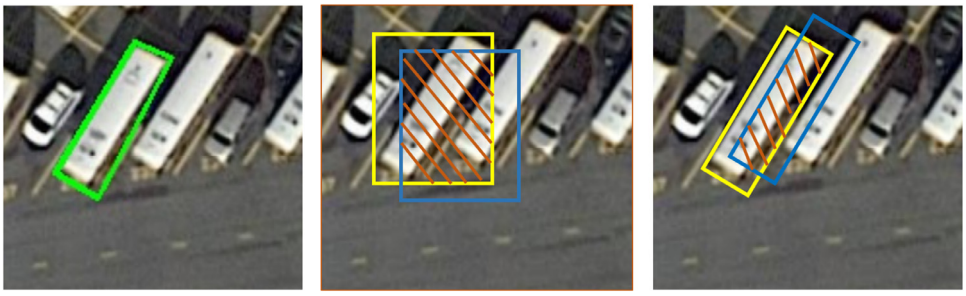


Fig. 6. The intersection over union (IoU) computation of RCR-CNN. From left to right: the quadrangle label, the calculated IoU on the HBB in the first R-CNN stage, and the calculated IoU on the rotated bounding box in the second R-CNN stage (the shaded part is the intersection area).

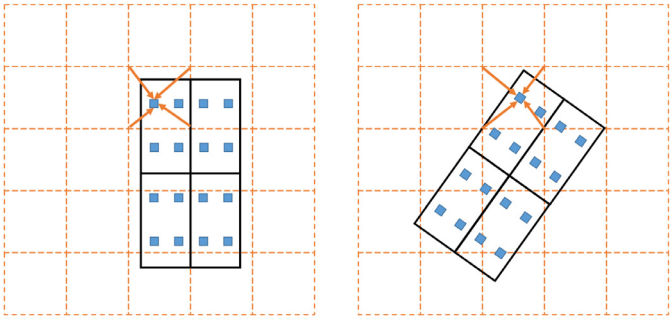


Fig. 7. RoIAlign and RRoIAlign.

Fig. 8. The coordinate system transformation in the RRoIAlign label calculation.

soccer ball field (SBF), swimming pool (SP), are labeled with quadrangles. The dataset is randomly divided into three parts; namely, 1/2 of the images in DOTA are the training set, 1/6 are the validation set, and 1/3 are the testing set. Some unclear objects are labeled as hard examples, which are defined as “do not care” in both the training stage and the testing stage.

We set hyperparameters following the mask R-CNN [7]. The backbone of our network is ResNet50 [51], which is pretrained on the ImageNet dataset. To avoid overfitting, we apply data augmentation for better performance. In particular, we rotate images with angles of 0, $\pi/2$, π, and $3\pi/2$, and we use class balance resampling to solve the class imbalance problem. In the DOTA experiment, we use only the DOTA training set to train our model. A stochastic gradient descent (SGD) optimizer is adopted to train the model. The momentum is 0.9, and the weight decay is 1×10^{-4}. The batch size is 1, and the number of iterations is 180,000. The learning rate is initialized as 2.5×10^{-3} and divided by 10 at the iteration range of (120000, 160000). All images are cropped to 1024×1024. We train and test the model with single scale input (1024×1024). We calculate the IoU of quadrangles for non-maximum suppression (NMS) as the default, and the IoU threshold is 0.3.

4.1.2. LocSLPR vs. PANet

We reimplement PANet [20] and use the segmentation result to generate the minimum rotated rectangle. The resolution of the PANet mask branch is 28×28. Correspondingly, we employ 28 sliding lines for LocSLPR in this comparative experiment. The instance

Fig. 9. An example comparison of the detection results for DOTA between LocSLPR (left) and direct regression (right).

Table 1
Ablation experiments on DOTA (in %).

Method	PANet [20]	Direct Regression	LocSLPR	LocSLPR+ RCR-CNN
Plane	89.48	88.92	89.61	89.93
BD	65.69	76.88	76.61	77.66
Bridge	47.74	42.13	40.93	44.01
GTF	63.56	57.18	63.55	62.77
SV	50.99	67.21	67.34	67.00
LV	57.95	64.98	67.60	69.80
Ship	69.86	78.36	78.62	79.57
TC	90.82	90.86	90.86	90.85
BC	70.75	79.22	79.63	80.19
ST	77.00	78.45	78.64	78.64
SBF	57.39	51.94	51.45	58.82
RA	54.67	56.25	63.42	56.50
Harbor	58.57	57.79	58.46	68.57
SP	55.99	61.87	62.05	63.64
HC	51.67	52.08	50.42	53.03
mAP	64.15	66.95	67.95	69.47
FPS	3.7	6.4	5.4	2.3

segmentation method, aiming to handle objects in any shape, can also be used for quadrangular object detection. However, the segmentation branch needs to increase the resolution of the mask map to generate more accurate quadrangles, which requires more parameters. Thus, for the quadrangular objects and curved polygon labeled text lines, it is not necessary to generate masks. In contrast, LocSLPR is a regression method that can regress in unlimited precision with limited parameters, and the smooth L_1 loss [50] is less sensitive to outliers. In Table 1, we compare our LocSLPR with PANet. Clearly, LocSLPR with a mAP of 67.95% significantly outperforms PANet with a mAP of 64.15%.

4.1.3. LocSLPR vs. direct regression

Direct regression the four vertices of a quadrangle leads to vertex order ambiguity in some cases. In our rule, the ambiguity will appear at 45°. In Table 1, we compare our LocSLPR with direct regression. We can observe that LocSLPR achieves better performance. The performance gain should be more significant if there are more samples with the 45° rotation. We show an example of detection results for DOTA in Fig. 9. Obviously, the direct regression approach seems to confuse which vertex is the top left point for those 45° rotated objects and generates the wrong coordinates. However, such problems can be well addressed in our proposed LocSLPR approach.

4.1.4. LocSLPR vs. LocSLPR+RCR-CNN

We used RCR-CNN because the target is a quadrangle. As shown in Table 1, the proposed two-stage approach, i.e., LocSLPR+RCR-CNN, achieves better mAP results than the one-stage LocSLPR approach. In the first stage, we propose horizontal rectangles. However, calculating the IoU of horizontal rectangles cannot always locate objects well. Specifically, when the objects have 45° rotation and are very dense, one proposal box may intersect with many objects with high IoU, leading to inaccurate detection results. Therefore, we use RCR-CNN and calculate the IoU of the rotated rectangle in the second stage. In addition, we regress again in the second stage, which can generate more precise quadrangles. An example of detection results for DOTA between LocSLPR and LocSLPR+RCR-CNN is illustrated in Fig. 10.

4.1.5. Efficiency

We compare the efficiency of our method with others. As shown in Table 1, the proposed LocSLPR is a little slower than direct regression, but compared with PANet which is an instance segmentation method, our method is more efficient. RCR-CNN is much slow because there are two R-CNNs in RCR-CNN, and R-CNN consumes much computation.

4.1.6. ICPR contest on object detection in aerial images

Based on our LocSLPR, we combine both the training set and the validation set of DOTA for training. To obtain the best results, we adopt ResNeXt-101 (32 × 8d) [52] as the backbone. Multi-scale testing is also used for Task 1 of the ICPR ODAI. This single model yields a 69.2% mAP. We also fuse this model with a segmentation-based model whose mAP is 67.5%. Finally, our best-submitted system for ODAI achieved a 70.5% mAP and was the champion system of the ODAI competition. Table 2 summarizes the entries from the Oriented Leaderboard on ODAI. It is worth mentioning that our USTC-NELSLIP system significantly outperforms other competitors, with an absolute gain of 8.3% mAP in comparison to the second-place system.

4.2. Text detection

4.2.1. Experiments on ICDAR2015 incidental scene text

The ICDAR2015 Incidental Scene Text dataset [53] is a commonly used benchmark for detecting arbitrary-angle quadrangular text lines. It contains 1000 images for training and 500 images for testing. The size of all images is 1280 × 720 pixels. Some words that are too small or unclear are annotated as “do not care” samples.

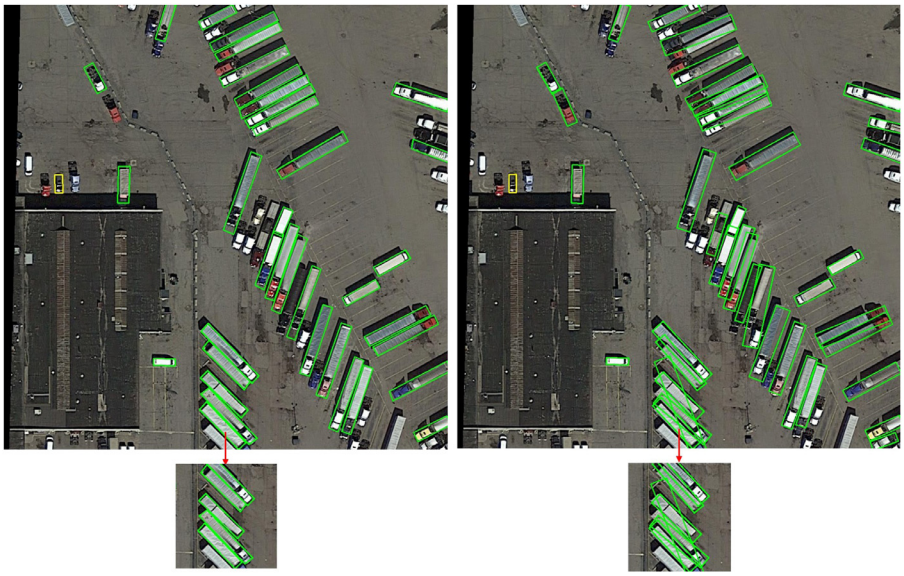


Fig. 10. An example comparison of detection results on DOTA between LocSLPR+RCR-CNN (left) and LocSLPR (right).

Table 2
Task1 - oriented leaderboard on ODAI.

Team name	USTC-NELSLIP (ours)	jmkoo	HUST_MCLAB	NWPU_SAIP	changzhonghan	madebyrag	mfhan
Plane	0.902	0.876	0.887	0.796	0.802	0.786	0.752
BD	0.852	0.677	0.754	0.680	0.449	0.745	0.527
Bridge	0.430	0.457	0.347	0.224	0.430	0.159	0.133
GTF	0.686	0.512	0.629	0.610	0.468	0.612	0.355
SV	0.740	0.705	0.585	0.602	0.665	0.343	0.549
LV	0.768	0.684	0.633	0.657	0.699	0.419	0.545
Ship	0.731	0.707	0.635	0.565	0.695	0.337	0.445
TC	0.900	0.904	0.901	0.887	0.896	0.874	0.771
BC	0.843	0.675	0.698	0.748	0.659	0.661	0.439
ST	0.761	0.669	0.688	0.666	0.695	0.530	0.491
SBF	0.639	0.400	0.508	0.610	0.438	0.544	0.177
RA	0.495	0.434	0.391	0.383	0.338	0.356	0.293
Harbor	0.556	0.556	0.471	0.425	0.456	0.379	0.306
SP	0.632	0.541	0.441	0.393	0.001	0.387	0.324
HC	0.639	0.534	0.405	0.420	0.274	0.461	0.194
mAP	0.705	0.622	0.598	0.578	0.531	0.506	0.420

Table 3
The performance comparison with other state-of-the-art methods on ICDAR2015 Incidental Scene Text dataset.

Methods	Precision (%)	Recall (%)	F-measure (%)	FPS
HUST [53]	44.0	37.8	40.7	–
Zhang et al. [54]	70.8	43.1	53.6	–
RRPN [23]	82.2	73.2	77.4	–
WordSup [29]	79.3	77.0	78.2	–
EAST [2]	83.3	78.3	80.7	–
Deep direct regression [1]	82.0	80.0	81.0	–
R^2CNN [24]	85.6	79.7	82.5	0.4
PixelLink [35]	85.5	82.0	83.7	3.0
FSTN [5]	**88.6**	80.0	84.1	–
Lyu *et al.* [37]	89.5	79.7	84.3	1
SLPR [49]	85.5	83.6	84.5	–
Textboxes++ [55]	91.2	79.2	84.8	–
LocSLPR+RCR-CNN	88.5	**86.2**	**87.3**	1.8

To evaluate our method on this dataset, we use ResNet50 as the backbone, which is pretrained on the ImageNet dataset. We combine the training dataset of the ICDAR2015 Incidental Scene Text with the training dataset of the ICDAR2013 competition [56] to train the model. Data augmentation is adopted for better performance. Specifically, we rotate images by $[0^\circ, 30^\circ, \ldots, 360^\circ]$ and randomly resize images to [600, 700, 800, 900, 1000, 1100]. We ignore the text lines that are labeled as "do not care" or whose short side is less than 10 pixels. We train the model by an SGD optimizer with the same parameter settings as in the DOTA experiments. In the inference phase, we resize the short side of the testing images to 1000, while keeping their aspect ratios unchanged. We compare our method with other state-of-the-art methods in Table 3. The LocSLPR+RCR-CNN method achieves an 87.3% F-measure with the

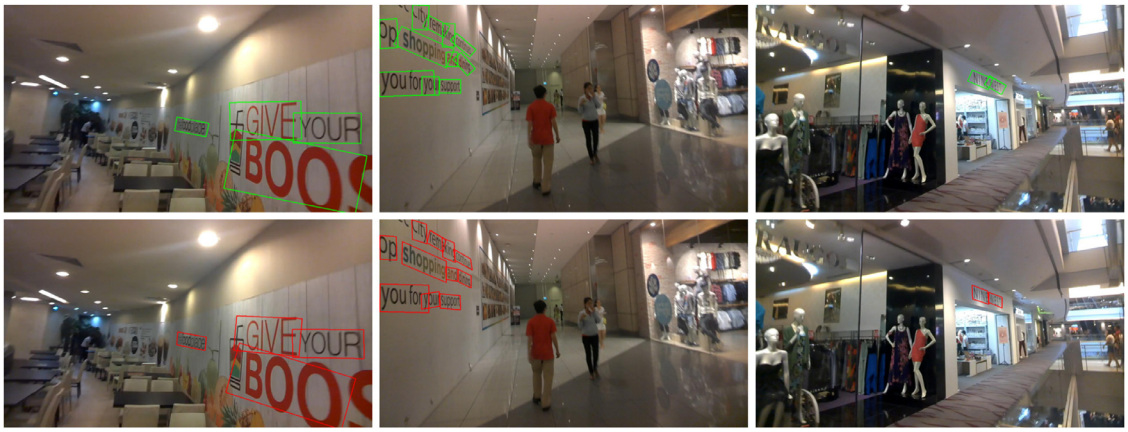


Fig. 11. An example comparison of detection results between the one-stage LocSLPR approach (top) and the two-stage LocSLPR+RCR-CNN approach (bottom) on the ICDAR2015 Incidental Scene Text dataset.

Fig. 12. An example comparison of detection results among different approaches on CTW1500 dataset. The green rectangle is the proposal box, while the yellow polygon is the final detection result. (From left to right: CTD+TLOC, SLPR, LocSLPR+RCR-CNN). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

single-scale input, significantly outperforming other state-of-the-art methods. We also show some examples of the detection results in Fig. 11. For the two-stage LocSLPR+RCR-CNN approach, LocSLPR is used to generate rotated rectangles in the first stage, while direct regression is adopted to generate quadrangles in the second stage. With the RCR-CNN, the detection results are obviously better than those obtained using only LocSLPR.

4.2.2. The thres of NMS

We set different threshold of IoU for NMS on ICDAR2015 and DOTA. As shown in 4, we find there are different optimal values in different datasets. NMS is not the main problem of our paper, so we set the threshold of IoU to 0.3 for consistency.

Table 4
Experiments on the threshold of NMS (in %).

Thres	0.1	0.3	0.5
ICDAR2015 (F-measure)	87.02	87.33	87.13
DOTA val (mAP)	67.45	67.41	68.05

4.2.3. Experiments on CTW1500

The curved text dataset (CTW1500) [10] is constructed by Yuliang *et al.* Different from traditional quadrangular text datasets, every text line in CTW1500 is labeled by a polygon with 14 points. In addition, the evaluation protocol calculates the IoU between polygons, which is specifically designed for curved texts.

Fig. 13. Examples of detection results using our LocSLPR+RCR-CNN method for the ICPR-MTWI challenge.

To evaluate our approach on curved texts, LocSLPR is used for both stages. We adopt 7 sliding lines for LocSLPR in the first stage but 28 sliding lines for LocSLPR in the second stage because we only need to generate rotated rectangles in the first stage, while the accurate contours of targets should be generated in the second stage. Thus, we can accelerate without performance degradation. The model is trained by an SGD optimizer and pretrained on the ImageNet dataset. The momentum and weight decay parameters are the same as those in the experiments on the ICDAR2015 Incidental Scene Text. The learning rate is initialized as 2.5×10^{-3}, which is divided by 10 in the iteration range (30000, 40000). The total number of iterations is 45,000. In the training stage, we only use the CTW1500 training set and resize the short side of the images to 600 without other data augmentation, which is the same as CTD+TLOC [10] and SLPR [49]. In the testing stage, we also resize the short side of the images to 600 and use the single-scale input. In order to speedup, we calculate the IoU of the rotated rectangles for NMS instead of the polygon, and the IoU threshold is 0.3. We compare our method with other state-of-the-art methods in Table 5. The proposed LocSLPR+RCR-CNN approach achieves an 83.1% F-measure on CTW1500, which significantly outperforms other methods and obtains an F-measure gain of 8.3% over our previous SLPR method [49]. This result shows that our method can handle the curved texts well. In Fig. 12, a qualitative comparison of the detection results on the CTW1500 dataset among different approaches is given. Compared with CTD+TLOC [10] and SLPR [49],

Table 5
A performance comparison between our method and other state-of-the-art methods on the CTW1500 dataset (* indicates the result is from Liu et al. [10]).

Method	Precision (%)	Recall (%)	F-measure (%)
Seglink* [28]	42.3	40.0	40.8
SWT* [57]	20.7	9.0	9.0
CTPN* [58]	60.4	53.8	56.9
EAST* [2]	78.7	49.1	60.4
DMPNet* [3]	69.9	56.0	62.2
CTD+TLOC [10]	77.4	69.8	73.4
CTD [10]	74.3	65.2	69.5
SLPR [49]	80.1	70.1	74.8
LocSLPR+RCR-CNN	**83.3**	**83.0**	**83.1**

Table 6
The leaderboard (Top-6) of the ICPR contest for robust reading for multi-type web images.

Team name	Score	Precision	Recall
nelslip(iflytek&ustc) (ours)	**0.796**	**0.813**	**0.779**
SRC-B-MachineLearningLab	0.766	**0.813**	0.723
UC	0.755	0.788	0.725
NTAI	0.752	0.799	0.711
NJUImagineLabPSENet	0.752	0.785	0.721
mclabdet	0.734	0.788	0.687

our LocSLPR+RCR-CNN approach generates more accurate polygon boxes, benefiting from RRoIAlign and LocSLPR.

4.2.4. Experiments on ICPR-MTWI challenge

In 2018, the ICPR Contest for Robust Reading for Multi-Type Web Images (ICPR-MTWI) [59] was held. The organizers provided 10,000 images for training and 10,000 images for testing. Different from previous datasets, these images were mainly collected from the Internet. Some blurred text lines are labeled as "do not care" samples. The evaluation protocol follows ICDAR2013 Born-Digital Image [56] but partially modifies the threshold.

To evaluate our method for the ICPR-MTWI challenge, we adopt ResNeXt-101 (32 $\times$ 8d) [52] as the backbone, which is pretrained on the ImageNet dataset. We randomly select 9000 images from the training set of ICPR-MTWI for training and use the remaining 1000 images for validation. We use data augmentation for better performance. Specifically, we rotate images by [0°, 90°, 180°, 360°] and randomly resize the short side of images to [700, 800, 900, 600, 500, 400]. We ignore the text lines that are labeled as "do not care". Unlike general object detection, some text lines can be very long, but the receptive field of CNN is limited; text lines that are too long or too short cannot be easily recognized at unsuitable resolution. Therefore, the text lines whose short side is less than 10 pixels or whose long side is longer than 612 pixels are also ignored. The model is trained by an SGD optimizer with the same parameter settings as in the DOTA experiments. To obtain the best performance, we keep the aspect ratio unchanged and resize the short side of the images to [400, 600, 800, 1000] and then evaluate our method using multi-scale inputs. We do not use model ensembling in this challenge. We show the leaderboard (Top-6 from more than 100 submitted systems) of the ICPR-MTWI challenge in Table 6. Our LocSLPR+RCR-CNN method with the team name "nelslip(iflytek&ustc)" achieved an F-score of 0.796, which was the best result among all submitted systems, yielding an absolute gain of 3% over the second-place system, with the team name "SRC-B-MachineLearningLab". This dataset contains text lines with arbitrary angles. As shown in Fig. 13, some watermark texts are very unclear, and some texts may intersect with other texts. All these problems increase the difficulty of this task. Clearly, our LocSLPR+RCR-CNN method addresses these issues quite well.

5. Conclusion

ODAI and text detection remain challenging due to complex background and large variations in the shape and size of objects. In this study, we present LocSLPR and RCR-CNN for shape-robust object detection. We prove that there is no ambiguity problem of vertex order in LocSLPR when we calculate the labels of regression. In addition, we analyze the reason why directly regressing four vertices is highly sensitive to labeling sequence, and we conduct experiments to support our viewpoint. Our method can achieve better performance with fewer parameters than segmentation-based method (PANet). We also verify that gradually regressing targets with RCR-CNN can generate more accurate results.

We perform experiments on many tasks for evaluation. Our method achieves state-of-the-art performance on DOTA and obtains better performance than PANet and direct regression methods. We also won the ICPR Contest on Object Detection in Aerial Images with great advantages. For text detection tasks, our method achieves state-of-the-art performance on the ICDAR2015 Incidental Scene Text dataset and CTW1500, yielding an F-score of 87.3% on ICDAR2015 and an F-score of 83.1% on CTW1500. In CTW1500, our method surpasses the second-best record by 8.3% in F-score. In addition, our method also won the ICPR Contest for Robust Reading for Multi-Type Web Images and surpassed the second competitor by 3.0% in F-score. All these results demonstrate the effectiveness of our method and show that our method is very versatile. The limitation of our method is that it can't achieve real-time detection compared with one-stage detector. In the future, we will explore efficient and accurate one-stage detector.

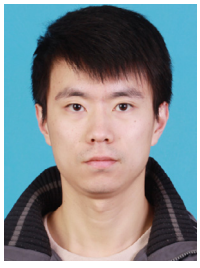
Acknowledgment

This work was supported in part by the National Key R&D Program of China under Grant 2017YFB1002202, in part by the National Natural Science Foundation of China under Grants 61671422 and U1613211, in part by the Key Science and Technology Project of Anhui Province under Grant 17030901005, and in part by the MOE-Microsoft Key Laboratory of University of Science and Technology of China.

References

[1] W. He, X.-Y. Zhang, F. Yin, C.-L. Liu, Deep direct regression for multi-oriented scene text detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 745–753.
[2] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, J. Liang, East: an efficient and accurate scene text detector, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5551–5560.
[3] Y. Liu, L. Jin, Deep matching prior network: Toward tighter multi-oriented text detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3454–3461.
[4] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, L. Zhang, Dota: a large-scale dataset for object detection in aerial images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018.
[5] Y. Dai, Z. Huang, Y. Gao, Y. Xu, K. Chen, J. Guo, W. Qiu, Fused text segmentation networks for multi-oriented scene text detection, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 3604–3609.
[6] Q. Yang, M. Cheng, W. Zhou, Y. Chen, M. Qiu, W. Lin, Inceptext: a new inception-text module with deformable psroi pooling for multi-oriented scene text detection.
[7] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.
[8] M. He, Y. Liu, Z. Yang, S. Zhang, C. Luo, F. Gao, Q. Zheng, Y. Wang, X. Zhang, L. Jin, Icpr2018 contest on robust reading for multi-type web images, in: International Conference on Pattern Recognition, IEEE, 2018, pp. 7–12.
[9] J. Ding, Z. Zhu, G.-S. Xia, X. Bai, S. Belongie, J. Luo, M. Datcu, M. Pelillo, L. Zhang, Icpr2018 contest on object detection in aerial images (odai-18), in: International Conference on Pattern Recognition, IEEE, 2018, pp. 1–6.
[10] Y. Liu, L. Jin, S. Zhang, C. Luo, S. Zhang, Curved scene text detection via transverse and longitudinal sequence connection, Pattern Recognit. 90 (2019) 337–345.
[11] 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.
[12] J. Dai, Y. Li, K. He, J. Sun, R-fcn: object detection via region-based fully convolutional networks, in: Advances in Neural Information Processing Systems, 2016, pp. 379–387.
[13] J. Wang, X. Tao, M. Xu, Y. Duan, J. Lu, Hierarchical objectness network for region proposal generation and object detection, Pattern Recognit. 83 (2018) 260–272.
[14] P. Zhang, W. Liu, Y. Lei, H. Lu, Hyperfusion-net: hyper-densely reflective feature fusion for salient object detection, Pattern Recognit. 93 (2019) 521–533.
[15] Q. Zhang, Z. Huo, Y. Liu, Y. Pan, C. Shan, J. Han, Salient object detection employing local tree-structured low-rank representation and foreground consistency, Pattern Recognit. (2019).
[16] F. Ghadiri, R. Bergevin, G.-A. Bilodeau, From superpixel to human shape modelling for carried object detection, Pattern Recognit. 89 (2019) 134–150.
[17] Z. Cai, N. Vasconcelos, Cascade r-cnn: delving into high quality object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6154–6162.

[18] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[19] 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, Springer, 2016, pp. 21–37.
[20] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8759–8768.
[21] P. Tang, X. Wang, Z. Huang, X. Bai, W. Liu, Deep patch learning for weakly supervised object classification and discovery, Pattern Recognit. 71 (2017) 446–459.
[22] P. Tang, X. Wang, S. Bai, W. Shen, X. Bai, W. Liu, A.L. Yuille, Pcl: proposal cluster learning for weakly supervised object detection, IEEE Trans. Pattern Anal. Mach. Intell. (2018).
[23] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, X. Xue, Arbitrary-oriented scene text detection via rotation proposals, IEEE Trans. Multimedia (2018).
[24] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, Z. Luo, R2cnn: rotational region cnn for orientation robust scene text detection, arXiv:1706.09579v1 (2017).
[25] A. Graves, A.-r. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, IEEE, 2013, pp. 6645–6649.
[26] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, X. Li, Single shot text detector with regional attention, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3047–3055.
[27] M. Liao, B. Shi, X. Bai, X. Wang, W. Liu, Textboxes: a fast text detector with a single deep neural network., in: AAAI, 2017, pp. 4161–4167.
[28] B. Shi, X. Bai, S. Belongie, Detecting oriented text in natural images by linking segments, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2550–2558.
[29] H. Hu, C. Zhang, Y. Luo, Y. Wang, J. Han, E. Ding, Wordsup: exploiting word annotations for character based text detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4940–4949.
[30] S. Tian, S. Lu, C. Li, Wetext: scene text detection under weak supervision, in: IEEE International Conference on Computer Vision, 2017.
[31] D. NguyenVan, S. Lu, S. Tian, N. Ouarti, M. Mokhtari, A pooling based scene text proposal technique for scene text reading in the wild, Pattern Recognit. 87 (2019) 118–129.
[32] M. Pastor, Text baseline detection, a single page trained system, Pattern Recognit. 94 (2019) 149–161.
[33] S. Roy, P. Shivakumara, H.A. Jalab, R.W. Ibrahim, U. Pal, T. Lu, Fractional poisson enhancement model for text detection and recognition in video frames, Pattern Recognit. 52 (2016) 433–447.
[34] Y. Wu, P. Natarajan, Self-organized text detection with minimal post-processing via border learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5000–5009.
[35] D. Deng, H. Liu, X. Li, D. Cai, Pixellink: detecting scene text via instance segmentation, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[36] D. He, X. Yang, C. Liang, Z. Zhou, G. Alexander, I. Ororbia, D. Kifer, C.L. Giles, Multi-scale fcn with cascaded instance aware segmentation for arbitrary oriented word spotting in the wild., in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 474–483.
[37] P. Lyu, C. Yao, W. Wu, S. Yan, X. Bai, Multi-oriented scene text detection via corner localization and region segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7553–7563.
[38] T. He, Z. Tian, W. Huang, C. Shen, Y. Qiao, C. Sun, An end-to-end textspotter with explicit alignment and attention, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5020–5029.
[39] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, J. Yan, Fots: fast oriented text spotting with a unified network, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5676–5685.
[40] Q. Jiang, L. Cao, M. Cheng, C. Wang, J. Li, Deep neural networks-based vehicle detection in satellite images, in: International Symposium on Bioelectronics and Bioinformatics (ISBB), IEEE, 2015, pp. 184–187.
[41] X. Chen, S. Xiang, C.-L. Liu, C.-H. Pan, Vehicle detection in satellite images by hybrid deep convolutional neural networks, IEEE Geosci. Remote Sens. Lett. 11 (10) (2014) 1797–1801.
[42] A.-B. Salberg, Detection of seals in remote sensing images using features extracted from deep convolutional neural networks, in: Geoscience and Remote Sensing Symposium, 2015, pp. 1893–1896.
[43] Y. Long, Y. Gong, Z. Xiao, Q. Liu, Accurate object localization in remote sensing images based on convolutional neural networks, IEEE Trans. Geosci. Remote Sens. 55 (5) (2017) 2486–2498.
[44] M.-R. Hsieh, Y.-L. Lin, W.H. Hsu, Drone-based object counting by spatially regularized regional proposal network, in: IEEE International Conference on Computer Vision, 1, 2017.
[45] Z. Zou, Z. Shi, Random access memories: a new paradigm for target detection in high resolution aerial remote sensing images, IEEE Trans. Image Process. 27 (3) (2018) 1100–1111.
[46] K. Li, G. Cheng, S. Bu, X. You, Rotation-insensitive and context-augmented object detection in remote sensing images, IEEE Trans. Geosci. Remote Sens. 56 (4) (2018) 2337–2348.
[47] G. Cheng, P. Zhou, J. Han, Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images, IEEE Trans. Geosci. Remote Sens. 54 (12) (2016) 7405–7415.
[48] G. Cheng, J. Han, P. Zhou, D. Xu, Learning rotation-invariant and fisher discriminative convolutional neural networks for object detection, IEEE Trans. Image Process. 28 (1) (2019) 265–278.
[49] Y. Zhu, J. Du, Sliding line point regression for shape robust scene text detection, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 3735–3740.
[50] R. Girshick, Fast r-cnn, in: IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.
[51] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[52] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2017, pp. 5987–5995.
[53] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V.R. Chandrasekhar, S. Lu, et al., Icdar 2015 competition on robust reading, in: International Conference on Document Analysis and Recognition, IEEE, 2015, pp. 1156–1160.
[54] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, X. Bai, Multi-oriented text detection with fully convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4159–4167.
[55] M. Liao, B. Shi, X. Bai, Textboxes++: a single-shot oriented scene text detector, IEEE Trans. Image Process. 27 (8) (2018) 3676–3690.
[56] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez, S. Robles, J. Mas, D. Fernandez, J. Almazan, L. de las Heras, Icdar 2013 robust reading competition, in: Proceedings of the International Conference of Document Analysis and Recognition, 2013, pp. 1115–1124.
[57] B. Epshtein, E. Ofek, Y. Wexler, Detecting text in natural scenes with stroke width transform, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 2963–2970.
[58] Z. Tian, W. Huang, T. He, P. He, Y. Qiao, Detecting text in natural image with connectionist text proposal network, in: European Conference on Computer Vision, Springer, 2016, pp. 56–72.
[59] Icpr-mtwi, (https://www.alibabacloud.com/zh/campaign/ICPR2018).



Yixing Zhu Yixing Zhu received a B.Eng. degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), in 2017. He is currently a Master's candidate at USTC. His current research area includes deep learning, OCR and object detection in aerial images.

Chixiang Ma Chixiang Ma received a B.Eng. degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), in 2017. He is currently a Ph.D. candidate at USTC. His current research area includes deep learning and OCR.

Jun Du received B.Eng. and Ph.D. degrees from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), in 2004 and 2009, respectively. From 2004 to 2009, he was with the iFlytek Speech Lab of USTC. During the above period, he worked as an Intern twice for 9 months at Microsoft Research Asia (MSRA), Beijing. In 2007, he also worked as a Research Assistant for 6 months in the Department of Computer Science at the University of Hong Kong. From July 2009 to June 2010, he worked at iFlytek Research on speech recognition. From July 2010 to January 2013, he joined MSRA as an Associate Researcher, working on handwriting recognition, OCR and speech recognition. Since February 2013, he has been with the National Engineering Laboratory for Speech and Language Information Processing (NEL-SLIP) of USTC.