2019 IEEE Winter Conference on Applications of Computer Vision

Mask R-CNN with Pyramid Attention Network for Scene Text Detection

Zhida Huang[1,3,†,*], Zhuoyao Zhong[2,3,†,*], Lei Sun[3], Qiang Huo[3]
[1]School of Software & Microelectronics, Peking University
[2]School of Electronic and Information Engineering, South China University of Technology
[3]Microsoft Research Asia
hzhida@pku.edu.cn, zhuoyao.zhong@gmail.com, {lsun, qianghuo}@microsoft.com

Abstract

In this paper, we present a new Mask R-CNN based text detection approach which can robustly detect multi-oriented and curved text from natural scene images in a unified manner. To enhance the feature representation ability of Mask R-CNN for text detection tasks, we propose to use the Pyramid Attention Network (PAN) as a new backbone network of Mask R-CNN. Experiments demonstrate that PAN can suppress false alarms caused by text-like backgrounds more effectively. Our proposed approach has achieved superior performance on both multi-oriented (ICDAR-2015, ICDAR-2017 MLT) and curved (SCUT-CTW1500) text detection benchmark tasks by only using single-scale and single-model testing.

1. Introduction

Scene text detection has drawn increasing attentions from the computer vision community [13, 20, 25, 33, 41] since it has a wide range of applications in document analysis, robot navigation, OCR translation, image retrieval and augmented reality. However, because of diverse text variabilities in colors, fonts, orientations, languages and scales, extremely complex and text-like backgrounds, as well as some distortions and artifacts caused by image capturing like non-uniform illumination, low contrast, low resolution and occlusion, text detection in natural scene images is still an unsolved problem.

Nowadays, with the astonishing development of deep learning, great progress has been made in this field. Lots of state-of-the-art convolutional neural network (CNN) based object detection and segmentation frameworks, such as Faster R-CNN [32], SSD [23] and FCN [27], have been borrowed to solve the text detection problem and substantially outperform traditional MSER [29] or SWT [7] based bottom-up text detection approaches. For example, some approaches [39, 38] formulate text detection as a semantic segmentation problem and employ an FCN to make a pixel-level text/non-text prediction, based on which a text saliency map can be generated. As only coarse text-blocks can be detected from the saliency map, complex post-processing steps are needed to extract accurate bounding boxes of text-lines. Unlike FCN-based methods, another category of methods treats text as a specific object and leverages effective object detection frameworks like R-CNN [9], Faster R-CNN [32], SSD [23], YOLO [31] and DenseBox [14] to detect words or text-lines from images directly. Although these approaches are composed of simpler pipelines, they still struggle with curved text detection. To solve this problem, some recent approaches like PixelLink [6], FTSN [5], and IncepText [37], propose to formulate text detection as an instance segmentation problem so that both straight text and curved text can be detected in a unified manner. Specifically, PixelLink proposes to detect text by linking pixels within the same text instances together, while FTSN and IncepText borrow the FCIS framework [19] to solve the text detection problem. Although promising results have been achieved, the used instance segmentation approaches have now been surpassed by the latest state-of-the-art Mask R-CNN approach on general instance segmentation tasks [11]. Therefore, it is straightforward to use Mask R-CNN to further improve the text detection performance.

In this paper, we present an effective Mask R-CNN based text detection approach which can detect multi-oriented and curved text from natural scene images in a unified manner. To enhance the feature representation ability of Mask R-CNN, we propose to use the Pyramid Attention Network (PAN) [18] as a new backbone network of Mask R-CNN. Experiments demonstrate that PAN can suppress false alarms caused by text-like backgrounds more effectively. Our proposed approach has achieved superior performance on both multi-oriented (ICDAR-2015 [17], ICDAR-

*Equal contribution. †This work was done when Zhida Huang and Zhuoyao Zhong were interns in Speech Group, Microsoft Research Asia, Beijing, China.

978-1-7281-1975-5/19/$31.00 ©2019 IEEE
DOI 10.1109/WACV.2019.00086

2017 MLT [30]) and curved (SCUT-CTW1500 [26]) text detection benchmark tasks.

2. Related work

In this section, we focus on reviewing recently proposed CNN based text detection approaches and recent developments in instance segmentation tasks.

2.1. Text Detection

State-of-the-art CNN based object detection and segmentation frameworks have been widely used to solve the text detection problem recently. Some of these methods [39, 38] borrow the idea of semantic segmentation and employ an FCN to make a pixel-level text/non-text prediction, which produces a text saliency map for text detection. However, only coarse text-blocks can be detected from this saliency map, so complex post-processing steps are needed to extract accurate bounding boxes of text-lines. Another category of methods [15, 10, 40, 20, 28, 25, 41, 13] treats text as a specific object and leverages state-of-the-art object detection frameworks to detect word or text-lines from images directly. Jaderberg et al. [15] adapted R-CNN for text detection, while its performance was limited by the traditional region proposal generation methods. Gupta et al. [10] borrowed the YOLO framework and employed a fully-convolutional regression network to perform text detection and bounding box regression at all locations and multiple scales of an image. Zhong et al. [40] and Liao et al. [20] employed the Faster R-CNN and SSD frameworks to solve the word-level horizontal text detection problem, respectively. In order to extend Faster R-CNN and SSD to multi-oriented text detection, Ma et al. [28] and Liu et al. [25] proposed quadrilateral anchors to hunt for inclined text proposals which could better fit the multi-oriented text instances. To overcome the inefficiency of anchor mechanism [13], Zhou et al. [41] and He et al. [13] borrowed the idea of DenseBox and used a one-stage FCN to output pixel-wise textness scores as well as the quadrilateral bounding boxes through all locations and scales of an image. Although these approaches are composed of simpler pipelines, they still struggle with curved text detection. Recently, instead of detecting the whole words or text-lines directly, Tian et al. [34] and Shi et al. [33] adopted object detection methods to detect text segments firstly, then grouped these text segments into words or lines with some simple text-line grouping algorithms or the learned linkage information, respectively. Intuitively, these methods can be applied for curved text detection, but they make the total text detection pipeline more sophisticated. Moreover, the segment grouping problem itself is a nontrivial problem, especially when the layout is complex, e.g., text with large character spacing, which will affect the text detection performance too. To overcome the above problems, some recent approaches propose to formulate text detection as an instance segmentation problem so that both straight text and curved text can be detected in a unified manner. Deng et al. [6] proposed to detect text by linking pixels within the same text instances together. Dai et al. [5] and Yang et al. [37] adopted the FCIS framework [19] to solve the text detection problem. In this paper, we borrowed Mask R-CNN, which is the latest state-of-the-art instance segmentation approach, to further enhance the text detection performance.

2.2. Instance Segmentation

Instance segmentation is a challenging task because it requires the correct detection of all objects in an image while also precisely segmenting each instance. Dai et al. [3] proposed a complex multiple-stage cascade that predicts segment proposals from bounding-box proposals, followed by classification. Later, Li et al. [19] combined the segment proposal system in [2] and R-FCN [4] for fully convolutional instance segmentation (FCIS). Although fast, FCIS exhibits systematic errors on overlapping instances and creates spurious edges [11]. More recently, Mask R-CNN [11] extended Faster R-CNN [32] by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. It introduced RoIAlign [11] to replace RoIPool [8] to fix the pixel misalignment and used ResneXt [36] as the base network. Moreover, it took advantage of Feature Pyramid Network (FPN [21]) to strengthen feature representation ability and partially eased the problem of small object detection. In this paper, to further enhance the feature representation ability of Mask R-CNN, we propose to incorporate the Pyramid Attention Network (PAN) [18] into the Mask R-CNN framework. Experiments demonstrate that PAN can suppress false alarms caused by text-like backgrounds more effectively.

3. Our Method

Our Mask R-CNN based text detection network is composed of four modules: 1) A PAN backbone network that is responsible for computing a multi-scale convolutional feature pyramid over a full image; 2) A region proposal network (RPN) that generates rectangular text proposals; 3) A Fast R-CNN detector that classifies extracted proposals and outputs the corresponding quadrilateral bounding boxes; 4) A mask prediction network that predicts text masks for input proposals. A schematic view of our text detection network is depicted in Fig. 1 and details are described in the following subsections.

3.1. Pyramid Attention Network

Recently, Li et al. [18] proposed a Pyramid Attention Network (PAN) that combines the attention mechanism and spatial pyramid to extract precise dense features for semantic segmentation tasks. It mainly consists of two mod-

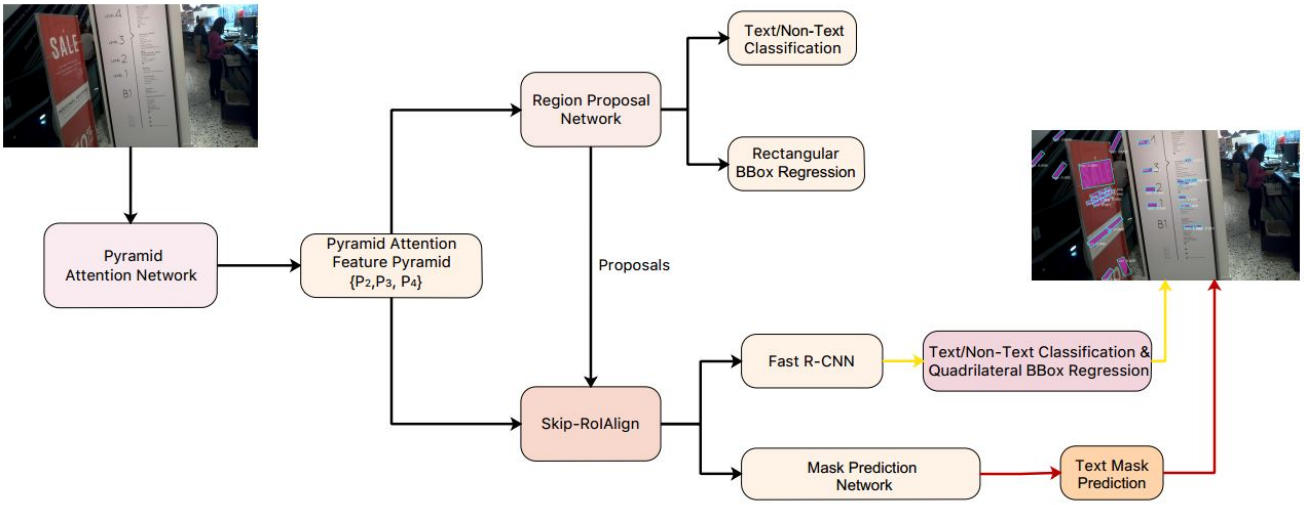

Figure 1: Architecture of our Mask R-CNN based text detector, which consists of a PAN backbone network, a region proposal network, a Fast R-CNN detector and a mask prediction network.

ules, i.e., a Feature Pyramid Attention (FPA) module and a Global Attention Up-sample (GAU) module. The FPA module performs spatial pyramid attention on high-level features and combines global pooling to learn better high-level feature representations. The GAU module is attached on each decoder layer to provide global context as a guidance of low-level features to select category localization details. Owing to these tactful designs, PAN achieves state-of-the-art segmentation performance on the VOC2012 and Cityscapes benchmark tasks. Inspired by this, we propose to use PAN as a new backbone network to improve the feature representation learning for our Mask R-CNN based text detection model.

We build PAN on top of ResNet50 [12] and ResNeXt50 [36]. The implementations of PAN generally follow [18] with just some modest modifications. As shown in Fig. 2, our FPA module takes the output features of the Res-4 layers in ResNet50 or ResNeXt50 as input, on which it performs 3×3 dilated convolution with sampling rates 3, 6, 12 respectively to better extract context information. These three feature maps are then concatenated and dimension reduced by a 1×1 convolution layer. After that, FPA performs a 1×1 convolution on the input Res-4 features further, whose output is multiplied with the above context features in a pixel-wise manner. The extracted features are added with the output features of the global pooling branch to get the final pyramid attention features. The GAU module, as is shown in Fig. 3, performs 3×3 convolution on the low-level features to reduce channels of feature maps from CNNs. The global context generated from high-level features is through a 1×1 convolution with instance normalization [35] and ReLU nonlinearity, then multiplied by the low-level features. Finally, the high-level features after up-sampling are added with the weighted low-level features to generate the GAU features. With the above FPA and GAU modules, we construct a powerful feature pyramid with three levels, i.e., P_2, P_3 and P_4, whose strides are 4, 8 and 16, respectively. The overall PAN architecture is depicted in Fig. 4. We refer readers to [18] for further details.

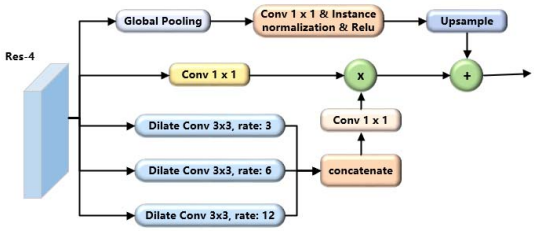

Figure 2: FPA of our PAN.

3.2. Region Proposal Network

Three RPNs are attached to P_2, P_3 and P_4 respectively, each of which slides a small network densely on the corresponding pyramid level to perform text/non-text classification and bounding box regression. The small network is implemented as a 3×3 convolutional layer followed by two sibling 1×1 convolutional layers, which are used for predicting textness score and rectangular bounding box locations respectively. As the size and aspect ratio variabilities

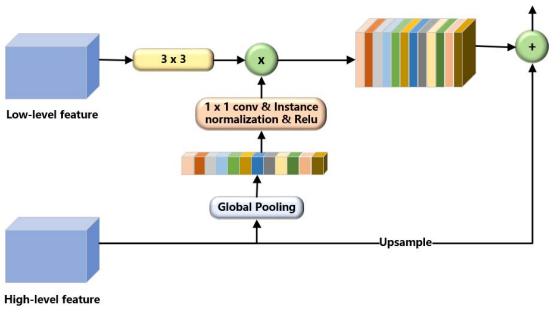

Figure 3: GAU of our PAN.

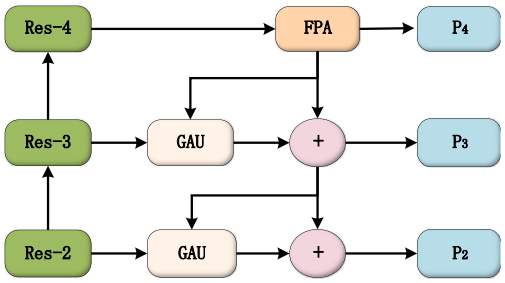

Figure 4: Architecture of our PAN.

of scene text instances are wider than general objects, we design a complicated set of anchors following [20]. Specifically, we design 6 anchors at each sliding position on each pyramid level in $\{P_2, P_3, P_4\}$ by using 6 aspect ratios $\{0.2, 0.5, 1.0, 2.0, 4.0, 8.0\}$ and one scale in $\{32, 64, 128\}$. The detection results of all three RPNs are aggregated together to construct a proposal set $\{D\}$. Then, we use the standard non-maximum suppression (NMS) algorithm with an IoU threshold of 0.7 to remove redundant proposals in $\{D\}$, and select the top-N scoring proposals for the succeeding Fast R-CNN and mask prediction network. N is set to 2000 in both the training and testing stages.

3.3. Fast R-CNN & Mask Prediction Network

After the region proposal generation step, extracting effective features for each proposal is critical to the performance of the following Fast R-CNN and mask prediction network. In the original Faster R-CNN [32], the features of all proposals are extracted from the last convolution layer of the backbone network, which would lead to insufficient features for small proposals. In the recent FPN [21], the features of proposals are extracted from different pyramid levels according to their sizes, i.e., the features of small proposals are extracted from low-level pyramid levels, while large proposals from high-level pyramid levels. Although more effective, there still exists room for further improvement

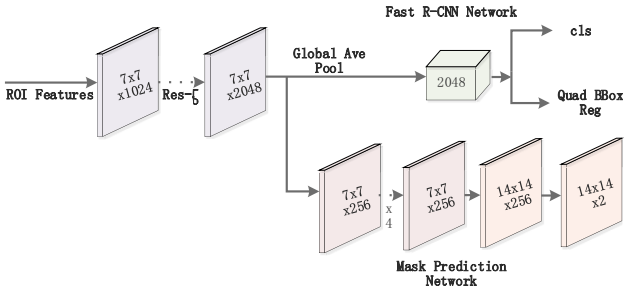

Figure 5: Architecture of Fast R-CNN and mask prediction network. The ROI features are first fed into the the 5-th stage of ResNet50 or ResneXt50, whose output features are shared by both Fast R-CNN and mask prediction network. Then, for Fast R-CNN, the output features are globally average pooled before the final text/non-text classification and quadrilateral bounding box regression layers, while for the mask prediction network, the output features are followed by four consecutive 3×3 convolutional layers and then upsampled before the final mask prediction layers. Numbers denote spatial resolution and channels.

[22]. As the features from the P_2 and P_3 levels have higher resolution and contain more detailed information, which are complementary to more abstract but low-resolution features from the P_4 level, it is straightforward to combine these three pyramid levels together to improve the feature representation ability. To achieve this, we borrow the idea of ION [1] and propose a Skip-RoIAlign method to fuse the P_2, P_3 and P_4 levels. Concretely, for each proposal, we apply ROIAlign over P_2, P_3 and P_4 pyramid levels respectively and extract three feature descriptors with a fixed spatial size of 7×7, which are concatenated and dimension reduced with a 1×1 convolutional layer to obtain the final ROI features. These ROI features are then fed into the network head for text/non-text classification, quadrilateral bounding box regression and mask prediction. Details of the network head are depicted in Fig. 5. The head includes the 5-th stage of ResNet50 or ResNeXt50, which is shared by the Fast R-CNN and mask prediction network.

3.4. Training

3.4.1 Loss Functions

Multi-task loss for RPN. There are two sibling output layers for each individual RPN, i.e., a text/non-text classification layer and a rectangular bounding box regression layer. The multi-task loss function can be denoted as follows:

$$L_{RPN_{p_i}} = L^R_{cls}(c, c^*) + \lambda_{loc} L^R_{loc}(r, r^*), \qquad (1)$$

where c and c^* are predicted and ground-truth labels respectively, $L^R_{cls}(c, c^*)$ is a softmax loss for classification

tasks; r and r^* represent the predicted and ground-truth 4-dimensional parameterized regression targets as stated in [32], $L^R_{loc}(r, r^*)$ is a smooth-L_1 loss [8] for regression tasks. λ_{loc} is a loss-balancing parameter, and we set $\lambda_{loc} = 3$.

The total loss of RPN L_{RPN} is the sum of the losses of the three RPNs.

Multi-task loss for Fast R-CNN. Fast R-CNN also has two sibling output layers: 1) A text/non-text classification layer, which is the same as the above-mentioned RPN; 2) A quadrilateral bounding box regression layer. The multi-task loss function for Fast R-CNN is defined as follows:

$$L_{FRCN} = L^F_{cls}(c, c^*) + \lambda_{loc} L^F_{loc}(t, t^*), \tag{2}$$

where $t = \{(\triangle_{x_i}, \triangle_{y_i}) | i \in \{1, 2, 3, 4\}\}$ and $t^* = \{(\triangle^*_{x_i}, \triangle^*_{y_i}) | i \in \{1, 2, 3, 4\}\}$ represent the predicted and ground-truth 8-dimensional parameterized coordinate offsets. Let $\{(x^g_i, y^g_i) | i \in \{1, 2, 3, 4\}\}$ denote the four vertices of G and $(x^p_1, y^p_1, x^p_2, y^p_2, P_w, P_h)$ be the top-left and bottom-right coordinates, width and height of an input proposal P. The parameterizations of t^* are denoted as:

$$\begin{aligned}
\triangle^*_{x_1} &= (x^g_1 - x^p_1)/P_w, \quad \triangle^*_{y_1} = (y^g_1 - y^p_1)/P_h, \\
\triangle^*_{x_2} &= (x^g_2 - x^p_2)/P_w, \quad \triangle^*_{y_2} = (y^g_2 - y^p_1)/P_h, \\
\triangle^*_{x_3} &= (x^g_3 - x^p_2)/P_w, \quad \triangle^*_{y_3} = (y^g_3 - y^p_2)/P_h, \\
\triangle^*_{x_4} &= (x^g_4 - x^p_1)/P_w, \quad \triangle^*_{y_4} = (y^g_4 - y^p_2)/P_h.
\end{aligned} \tag{3}$$

$L^F_{loc}(t, t^*)$ is also a smooth-L_1 loss and we set $\lambda_{loc} = 1$.

Loss for mask prediction network. Let m and m^* be the predicted and ground-truth mask targets respectively and $L_{mask}(m, m^*)$ be a standard binary cross-entropy loss for mask prediction tasks. Based on these definitions, the loss function can be defined as follows:

$$L_{MASK} = L_{mask}(m, m^*). \tag{4}$$

The overall loss function for training the proposed Mask R-CNN based text detection model can be denoted as:

$$L = L_{RPN} + L_{FRCN} + \lambda_{mask} L_{MASK}, \tag{5}$$

where λ_{mask} is a loss-balancing parameter for L_{MASK}, and we set $\lambda_{mask} = 0.03125$.

3.4.2 Training Details

In each training iteration of RPN, we sample a mini-batch of 128 positive and 128 negative anchors for each RPN. An anchor is assigned a positive label if it has the highest IoU for a given ground-truth bounding box or has an IoU over 0.7 with any ground-truth bounding box, and a negative label if its IoU overlap is less than 0.3 for all ground-truth bounding boxes. For Fast R-CNN, we sample a mini-batch of 64 positive and 192 negative text proposals in each iteration. A proposal is assigned a positive label if it has an IoU over 0.5 with any ground-truth bounding box, otherwise assigned a negative label. For the sake of efficiency, the IoU overlaps between proposals and ground-truth boxes are calculated using their axis-aligned rectangular bounding boxes. Only the positive text proposals are used for training the mask prediction network. The mask target is the intersection between a proposal and its associated ground-truth mask.

4. Experiments

We evaluate our proposed method on several standard benchmark tasks including ICDAR-2015 [17] and ICDAR-2017 MLT 2017 [30] for multi-oriented text detection, and SCUT-CTW 1500 [26] for curved text detection. Text instances are labeled in word-level with quadrilateral bounding boxes in the former two datasets and in text-line level with 14 coordinate points in SCUT-CTW 1500. ICDAR-2017 MLT is built for the multi-lingual scene text detection and script identification challenge in the ICDAR-2017 Robust Reading Competition, which includes 9 languages: Chinese, Japanese, Korean, English, French, Arabic, Italian, German and Indian. It contains 7,200, 1,800 and 9,000 images for training, validation and testing, respectively. ICDAR-2015 is built for the Incidental Scene Text challenge in the ICDAR-2015 Robust Reading Competition, which contains 1,000 and 500 images for training and testing. SCUT-CTW 1500 is a curved text detection dataset, including 1,000 training images and 500 testing images.

To make our results comparable to others, we use the online official evaluation tools to evaluate the performance of our approach on ICDAR-2017 MLT and ICDAR-2015, and use the evaluation tool provided by the authors of [26] on SCUT-CTW 1500.

4.1. Implementation Details

The weights of ResNet50 or ResNeXt50 related layers in the PAN backbone network are initialized by using the corresponding pre-trained models from the ImageNet classification task [12, 36]. The weights of the new layers for PAN, RPN, Fast R-CNN and mask prediction network are initialized by using random weights with a Gaussian distribution of mean 0 and standard deviation 0.01. Our Mask R-CNN based text detection model is trained in an end-to-end manner and optimized by the standard SGD algorithm with a momentum of 0.9 and weight decay of 0.0005. For ICDAR-2017 MLT, we use the training and validation data, i.e., a total of 9,000 images for training, while for both ICDAR-2015 and SCUT-CTW 1500, we only use the provided training images for training.

We implement our approach based on MXNet and experiments are conducted on a workstation with 4 Nvidia P100 GPUs. We adopt a multi-scale training strategy. The scale S is defined as the length of the shorter side of an image. In

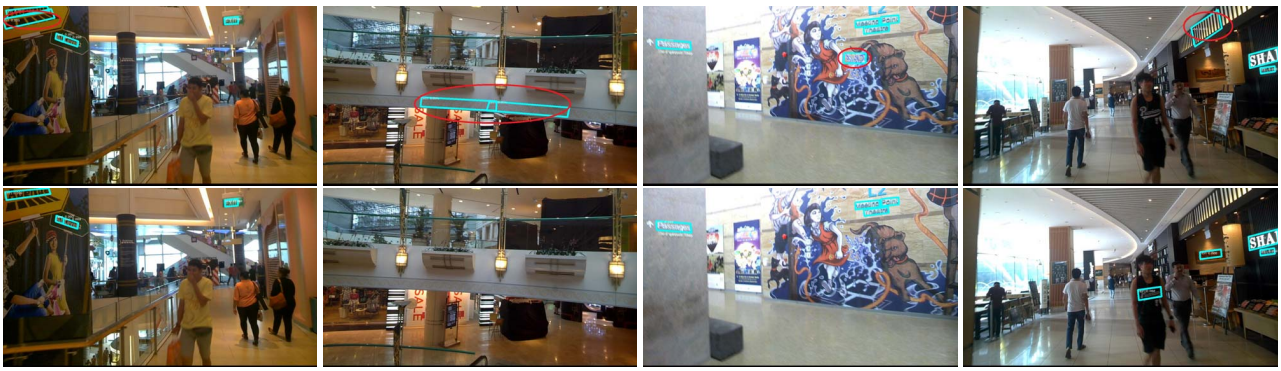

Figure 6: PAN can effectively suppress some false alarms caused by text-like backgrounds. The first row: detection results of Mask R-CNN with FPN. The second row: detection results of Mask R-CNN with PAN.

each training iteration, a selected training image is individually rescaled by randomly sampling a scale S from the set $\{480, 576, 720, 928, 1088\}$, $\{480, 576, 688, 720, 928\}$, and $\{300, 400, 500, 600, 704\}$ for ICDAR-2017 MLT, ICDAR-2015 and SCUT-CTW 1500, respectively.

In the testing phase, we keep the top-2000 scoring text proposals generated by RPN for the succeeding Fast R-CNN. After the Fast R-CNN step, quadrilateral bounding boxes of detected text instances are predicted and suppressed by the Skewed NMS [28] algorithm with an IoU threshold of 0.3. Finally, ROI features in the axis-aligned rectangular bounding box of each remaining text instance are fed into the mask prediction network to get the text mask. For the ICDAR-2017 MLT and ICDAR-2015 datasets, we directly use the quadrilateral bounding boxes predicted by the Fast R-CNN module as the final detection results, while for the curved text detection dataset SCUT-CTW 1500, we use the text masks predicted by the mask prediction network as the final detection results.

4.2. Component evaluation

In this section, we conduct a series of ablation experiments to evaluate the effectiveness of the base convolutional network and PAN on ICDAR-2017 MLT, ICDAR-2015 and SCUT-CTW 1500 text detection benchmark datasets. All the experiments are based on single-model and single-scale testing. The scales of testing images are set as 1440, 1024 and 512 for ICDAR-2017 MLT, ICDAR-2015 and SCUT-CTW 1500, respectively.

ResNeXt50 is better than ResNet50. As an important part of a backbone network (e.g., ResNet50-FPN), the base convolutional network (e.g., ResNet50) affects the text detection performance a lot. Here we compare the performance of two different base convolutional networks, i.e., Resnet50 and ResneXt50, on ICDAR-2017 MLT and ICDAR-2015. As shown in Table 1 and Table 2, ResneXt50 can consistently outperform ResNet50. In the following experiments, we will use ResneXt50 as our base convolutional network.

Base Network	FPN	PAN	R	P	F
ResNet50	✓		**0.687**	0.744	0.714
ResNet50		✓	0.686	**0.787**	**0.733**
ResneXt50	✓		0.686	0.795	0.737
ResneXt50		✓	**0.698**	**0.800**	**0.743**

Table 1: Component evaluation on ICDAR-2017 MLT. R, P and F stand for recall, precision and F-measure respectively.

Base Network	FPN	PAN	R	P	F
ResNet50	✓		**0.818**	0.877	0.846
ResNet50		✓	**0.818**	**0.882**	**0.849**
ResneXt50	✓		0.806	0.899	0.850
ResneXt50		✓	**0.815**	**0.908**	**0.859**

Table 2: Component evaluation on ICDAR-2015. R, P and F stand for recall, precision and F-measure respectively.

Base Network	FPN	PAN	R	P	F
ResneXt50	✓		0.826	0.839	0.833
ResneXt50		✓	**0.832**	**0.868**	**0.850**

Table 3: Component evaluation on SCUT-CTW 1500. R, P and F stand for recall, precision and F-measure respectively.

PAN is more powerful than FPN. We compare PAN with FPN [21] on all three datasets. As shown in Tables 1-3, no matter which base network is used, PAN consistently outperforms FPN on all datasets, which can demonstrate the effectiveness of PAN. The major improvement of PAN comes from the higher precision especially when the base network is relatively weaker, e.g., when ResNet50 is used as the base network on ICDAR-2017 MLT, PAN can increase the precision by 4.3% absolutely (Table 1). Some qualitative comparison examples on ICDAR-2015 are shown Fig. 6, from which we can find that some false alarms caused by text-like

Method	R	P	F
Proposed	**0.698**	**0.800**	**0.743**
FOTS MS [24]	0.623	0.818	0.707
SCUT DLVClab1 [30]	0.545	0.802	0.649
SARI FDU RRPN v1 [28]	0.555	71.17	0.623
TDN SJTU2017 [30]	0.471	0.642	0.543

Table 4: Comparison with prior arts on ICDAR-2017 MLT. R, P and F stand for recall, precision and F-measure respectively. MS indicates using multi-scale testing.

backgrounds could be suppressed by PAN.

4.3. Comparison with Prior Arts

We compare the performance of our approach with other most competitive results on the ICDAR-2017 MLT, ICDAR-2015 and SCUT-CTW 1500 text detection benchmark datasets. For fair comparisons, we report all results without using recognition information. As shown in Tables 4-6, our approach achieves the best performance on these three datasets by only using single-scale and single-model testing. Specifically, as shown in Table 4, our approach outperforms the most closest method [24] significantly by improving the F-measure from 0.707 to 0.743 on the challenging ICDAR-2017 MLT dataset, even though [24] applies multi-scale testing to achieve the best possible performance. On the ICDAR-2015 dataset, as shown in Table 5, even though some other approaches have used extra training data, our approach still achieves the best result of 0.815, 0.908 and 0.859 in recall, precision and F-measure respectively. On the SCUT-CTW 1500 dataset, our approach has achieved a new state-of-the-art result, i.e., 0.832, 0.868 and 0.850 in recall, precision and F-measure respectively as shown in Table 6, outperforming other methods by a large margin. The superior performance achieved by our proposed approach on these three challenging text detection benchmarks can demonstrate the advantage of our approach. Some qualitative detection results are depicted in Figs. 6-8.

5. Conclusion and Discussion

A new Mask R-CNN based text detection approach has been proposed in this paper. Thanks to the flexibility of Mask R-CNN, the proposed approach can detect multi-oriented and curved text from natural scene images robustly in a unified manner. Moreover, we demonstrate that using the Pyramid Attention Network (PAN) as a new backbone network of Mask R-CNN enhances the feature representation ability of Mask R-CNN significantly, so that false alarms caused by text-like backgrounds are suppressed more effectively. Our proposed approach has achieved superior performance on both multi-oriented (ICDAR-2015, ICDAR-2017 MLT) and curved (SCUT-CTW1500) text detection benchmark tasks by only using single-scale and single-model testing. However, our approach still has some limitations. First, the running speed of our approach is not fast enough due to the computation intensive PAN backbone network and Mask R-CNN framework. Moreover, our approach struggles with skewed nearby long text-lines owing to the limitation of rectangular proposals generated by RPN. More researches are needed to address these challenging problems.

Method	ExtraData	R	P	F
Proposed	×	**0.815**	**0.908**	**0.859**
IncepText [37]	×	0.806	0.905	0.853
FTSN [5]	✓	0.800	0.886	0.841
R2CNN [16]	✓	0.797	0.856	0.825
DDR [13]	✓	0.800	0.820	0.810
EAST [41]	–	0.783	0.832	0.807
RRPN [28]	✓	0.732	0.822	0.774
SegLink [33]	✓	0.731	0.768	0.749

Table 5: Comparison with prior arts on ICDAR-2015. R, P and F stand for recall, precision and F-measure respectively.

Method	R	P	F
Proposed	**0.832**	**0.868**	**0.850**
CTD+TLOC [26]	0.698	0.774	0.734
DMPNet [25]	0.560	0.699	0.622
EAST [41]	0.491	0.787	0.604
CTPN [34]	0.538	0.604	0.569

Table 6: Comparison with prior arts on SCUT-CTW 1500. R, P and F stand for recall, precision and F-measure respectively.

References

[1] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick. Inside-outside net: Detecting objects in context with skip pooling. In *CVPR*, pages 2874–2883, 2016.

[2] J. Dai, K. He, Y. Li, S. Ren, and J. Sun. Instance-sensitive fully convolutional networks. In *ECCV*, pages 534–549, 2016.

[3] J. Dai, K. He, and J. Sun. Instance-aware semantic segmentation via multi-task network cascades. In *CVPR*, pages 3150–3158, 2016.

[4] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In *NIPS*, pages 379–387, 2016.

[5] Y. Dai, Z. Huang, Y. Gao, and K. Chen. Fused text segmentation networks for multi-oriented scene text detection. *arXiv preprint arXiv:1709.03272*, 2017.

Figure 7: Detection results of our proposed Mask R-CNN based text detector on ICDAR-2017 MLT.

Figure 8: Detection results of our proposed Mask R-CNN based text detector on SCUT-CTW1500.

[6] D. Deng, H. Liu, X. Li, and D. Cai. Pixellink: Detecting scene text via instance segmentation. In *AAAI*, pages 6773–6780, 2018.

[7] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *CVPR*, pages 2963–2970, 2010.

[8] R. Girshick. Fast r-cnn. In *ICCV*, pages 1440–1448, 2015.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.

[10] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localisation in natural images. In *CVPR*, pages 2315–2324, 2016.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, pages 2980–2988, 2017.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[13] W. He, X. Zhang, F. Yin, and C. Liu. Deep direct regression for multi-oriented scene text detection. In *ICCV*, pages 745–753, 2017.

[14] L. Huang, Y. Yang, Y. Deng, and Y. Yu. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*, 2015.

[15] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 116(1):1–20, 2016.

[16] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, and Z. Luo. R2cnn: rotational region cnn for orientation robust scene text detection. *arXiv preprint arXiv:1706.09579*, 2017.

[17] 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 *ICDAR*, pages 1156–1160, 2015.

[18] H. Li, P. Xiong, J. An, and L. Wang. Pyramid attention network for semantic segmentation. In *BMVC*, 2018.

[19] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. *arXiv preprint arXiv:1611.07709*, 2016.

[20] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu. Textboxes: A fast text detector with a single deep neural network. In *AAAI*, pages 4161–4167, 2017.

[21] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944, 2017.

[22] S. Liu, L. Q. Zitnick, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *CVPR*, pages 8759–8768, 2018.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016.

[24] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan. Fots: Fast oriented text spotting with a unified network. In *CVPR*, pages 5676–5685, 2018.

[25] Y. Liu and L. Jin. Deep matching prior network: Toward tighter multi-oriented text detection. In *CVPR*, pages 3454–3461, 2017.

[26] Y. Liu, L. Jin, S. Zhang, and S. Zhang. Detecting curve text in the wild: New dataset and new solution. *CoRR*, abs/1712.02170, 2017.

[27] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.

[28] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Transactions on Multimedia*, 20(11):3111–3122, 2018.

[29] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.

[30] N. Nayef, F. Yin, I. Bizid, H. Choi, Y. Feng, D. Karatzas, Z. Luo, U. Pal, C. Rigaud, and J. Chazalon. Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification - rrc-mlt. In *ICDAR*, pages 1454–1459, 2018.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016.

[32] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[33] B. Shi, X. Bai, and S. J. Belongie. Detecting oriented text in natural images by linking segments. In *CVPR*, pages 3482–3490, 2017.

[34] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao. Detecting text in natural image with connectionist text proposal network. In *ECCV*, pages 56–72, 2016.

[35] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016.

[36] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *CVPR*, pages 5987–5995, 2017.

[37] Q. Yang, M. Cheng, W. Zhou, Y. Chen, M. Qiu, and W. Lin. Inceptext: A new inception-text module with deformable psroi pooling for multi-oriented scene text detection. *arXiv preprint arXiv:1805.01167*, 2018.

[38] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao. Scene text detection via holistic, multi-channel prediction. *CoRR*, abs/1606.09002, 2016.

[39] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-oriented text detection with fully convolutional networks. In *CVPR*, pages 4159–4167, 2016.

[40] Z. Zhong, L. Jin, and S. Huang. Deeptext: A new approach for text proposal generation and text detection in natural images. In *ICASSP*, pages 1–18, 2017.

[41] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. East: An efficient and accurate scene text detector. In *CVPR*, pages 2642–2651, 2017.