Exploring the Capacity of an Orderless Box Discretization Network for Multi-orientation Scene Text Detection

Yuliang Liu[1,2] · Tong He[2] · Hao Chen[2] · Xinyu Wang[2] · Canjie Luo[1] · Shuaitao Zhang[1] · Chunhua Shen[2,3*] · Lianwen Jin[1*]

Abstract Multi-orientation scene text detection has recently gained significant research attention. Previous methods directly predict words or text lines, typically by using quadrilateral shapes. However, many of these methods neglect the significance of consistent labeling, which is important for maintaining a stable training process, especially when it comprises a large amount of data. Here we solve this problem by proposing a new method, Orderless Box Discretization (OBD), which first discretizes the quadrilateral box into several key edges containing all potential horizontal and vertical positions. To decode accurate vertex positions, a simple yet effective matching procedure is proposed for reconstructing the quadrilateral bounding boxes. Our method solves the ambiguity issue, which has a significant impact on the learning process. Extensive ablation studies are conducted to validate the effectiveness of our proposed method quantitatively. More importantly, based on OBD, we provide a detailed analysis of the impact of a collection of refinements, which may inspire others to build state-of-the-art text detectors. Combining both OBD and these useful refinements, we achieve state-of-the-art performance on various benchmarks, including ICDAR 2015 and MLT. Our method also won the first place in the text detection task at the recent *ICDAR2019 Robust Reading Challenge for Reading Chinese Text on Signboards*, further demonstrating its superior performance. The code is available at https://git.io/TextDet.
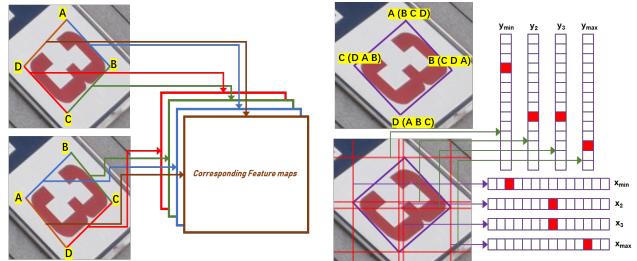
Keywords Scene text · Text detection · Orderless Box Discretization

[1]South China University of Technology, China
[2]The University of Adelaide, Australia
[3]Monash University, Australia
*Corresponding authors.

(a) Previous regression-based methods.   (b) Our proposed OBD.

Fig. 1: Comparison of (a) previous methods and (b) our proposed OBD. Previous methods directly regress the vertices, which can often be adversely affected by inconsistent labeling of training data, resulting in unstable training and unsatisfactory performances. Our method tackles this problem and removes the ambiguity by discretizing a quadrilateral bounding box that is orderless.

1 Introduction

Scene text detection in arbitrary orientations has garnered significant attention in computer vision because of its numerous potential applications, including augmented reality and robot navigation. Scene text detection is also the foundation and prerequisite for text recognition, which provides a reliable and straightforward approach to scene understanding. However, this challenge remains largely unsolved because text instances in natural images are often of multi-orientation, low-quality representations, having perspective distortions of various sizes and scales.

In the literature, several methods [1, 2, 3, 4, 5, 6] have been developed for solving horizontal scene text detection. However, scene text in the wild is typically presented in a multi-orientation form, attracting a few recent studies [7, 8, 9, 10, 10, 11, 12, 13, 14, 15, 16, 17, 18] that can be roughly categorized into two groups: segmentation and regression-based methods. Segmentation-based methods often employ networks, such as fully convolution networks

(FCNs) [19] and Mask R-CNN [20]. Segmentation-based methods have become the mainstream approach, because they are sufficiently robust in many complicated scenarios. One limitation is that segmented text instances often require additional post-processing steps. For example, the segmentation results obtained by Mask R-CNN must be fitted into rotated quadrilateral bounding boxes, which necessitates a number of heuristic settings and geometric assumptions.

On the other hand, Regression-based methods [8, 14, 15, 18, 21, 22, 23, 24, 25] are comparatively simple. For multi-orientation text, explicitly predicting the vertices obtains the four boundaries of the text instances. Thus, no additional grouping procedure is required. Although these methods can directly predict vertex positions, the significance of regression without facing inconsistent labeling has rarely been discussed. Consider the efficient and accurate scene text (EAST) detector [25] method as an example. In EAST, each feature within a text instance is responsible for regressing the corresponding quadrilateral bounding box by predicting four distances to the boundaries and a rotation angle from the viewpoint. A pre-processing step to assign regression targets is required. As shown in Figure 1, the regression targets can be altered drastically, even with a minor rotation. Such ambiguities lead to an unstable training process, which considerably degrades the performance. Our experiments indicate that the accuracy of EAST [25] deteriorates sharply (by more than 10%) when equipped with a random rotation technique for data augmentation, which is supposed to boost the performance.

To address this problem, we propose a novel method, (*i.e.*, Orderless Box Discretization (OBD)), which consists of two modules: *Key Edges Detection* and *Matching-Type Learning*. The fundamental idea is to employ invariant representations (*e.g.*, minimum *x*, minimum *y*, maximum *x*, maximum *y*, mean center point, and intersecting point of the diagonals) that are irrelevant to the label sequence to deduce the bounding box coordinates inversely. To simplify the parameterization, the OBD method first locates all discretized horizontal and vertical edges that contain a vertex. Then, a sequence labeling matching type is learned to determine the best-fit quadrilateral. By avoiding the ambiguity of the training targets, our approach successfully improves performance when a large amount of rotated data is involved.

We complement our method with a few critical technical innovations that further enhance performance. We conduct extensive experiments and ablation studies based on our method to explore the influence of six relevant issues: (namely, data arrangement, pre-processing, backbone, proposal generation, prediction head, and post-processing) to determine the significance of the various components. We thus provide useful tips for designing state-of-the-art text detectors. Leveraging OBD and these useful refinements, we won first place in the task of Text Line Detection at the *IC-*

DAR2019 Robust Reading Challenge on Reading Chinese Text on Signboards.

Our main contributions are summarized as follows.

1. Our method addresses the inconsistent labeling issue of regression-based methods, which is of great importance for achieving good detection accuracy.
2. The flexibility of our proposed method allows us to make use of several key refinements that are critical to further boosting accuracy. Our method achieves state-of-the-art performance on various scene text detection benchmarks, including ICDAR2015 [26] and MLT [27]. Additionally, our method won the first place in the Text Detection task of the recent *ICDAR2019 Robust Reading Challenge on Reading Chinese Text on Signboard*. Based on the detection results, we integrate advanced recognition models to achieve state-of-the-art results.
3. Our method can be generalized to ship detection in aerial images without minimum modification. The significant improvement in terms of the TIoU-Hmean metric further demonstrates the robustness of our approach.
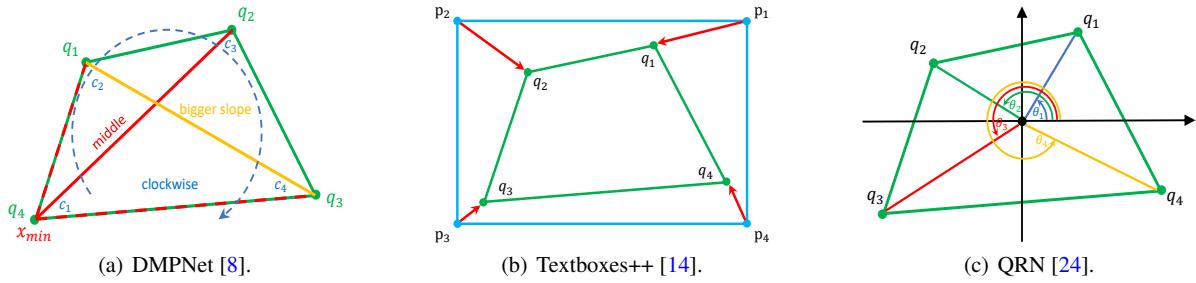
2 Related Work

Recently, the emergence of new datasets [28, 29, 30, 31] has propelled arbitrarily shaped scene text detection to mainstream research. Multi-orientation scene text detection is one of its most important representations, because multi-orientation scene text comprises most of the text found in real-world visual scenes. The computer-driven detection task remains complex, and there is much room for improvement with regards to decoding multi-orientation text from pictures. Hence, detection benchmarks, such as the MLT [27, 32] dataset, are leveraged to refine the process. However, using quadrilateral bounding boxes can result in some problems for both current segmentation and non-segmentation-based methods.

Segmentation-based. Segmentation-based methods [19, 20, 33, 34, 35, 36, 37, 38] usually require additional steps to group pixels into polygons.

Non-segmentation-based. Non-segmentation based methods [8, 14, 15, 18, 21, 22, 23, 24, 25] can directly learn the exact bounding box for localizing the text instances, but they are easily affected by the label sequence. Usually, such methods use a typical sorting method of the coordinate sequence to alleviate this issue. However, the solutions are not robust because the entire sequence may change even with a small amount of interference. To clarify this, we discuss some of the previous solutions as follows:

- Given an annotation having coordinates of four points, a common sorting method of the coordinate sequence

(a) DMPNet [8].   (b) Textboxes++ [14].   (c) QRN [24].

Fig. 2: Previous solutions can be negatively affected by the inconsistent labeling issue.

to alleviate this issue is to choose the point having the minimum x as the first point, then deciding the rest of the points in a clockwise manner. However, this protocol is not robust. Considering the horizontal rectangle as an example, using this protocol, let us decide that the first point is the top-left point. Thus, the fourth point is the bottom-left point. Suppose that the bottom-left point moves leftward one pixel (which is possible because of the inconsistent labeling). In that case, the original fourth point becomes the first point, and the whole sequence changes, resulting in very unstable learning.

- As shown in Figure 2(a), DMPNet [8] proposed a protocol that uses the slope to determine the sequence. However, if the diagonal is vertical, leftward, or rightward, change of a pixel can result in a completely different sequence.

- As shown in Figure 2(b), given four points, Textboxes++ [14] uses the distances between the annotation points and the vertices of the circumscribed horizontal rectangle to determine the sequence. However, if q_1 and q_4 have the same distance to p_1, and one pixel rotation can completely change the whole sequence.

- As shown in Figure 2(c), QRN [24] first finds the mean center point of the four given points then constructs a Cartesian coordinate system. Using the positive x axis, QRN ranks the intersection angles of the four points and chooses the point having the minimum angle as the first. However, if the first point is in the positive x axis, one pixel change upward or downward will result in an entirely different sequence.

Although these methods [8, 14, 24] can alleviate confusion to some extent, the results can be significantly undermined when using pseudo samples having large degrees of rotation.
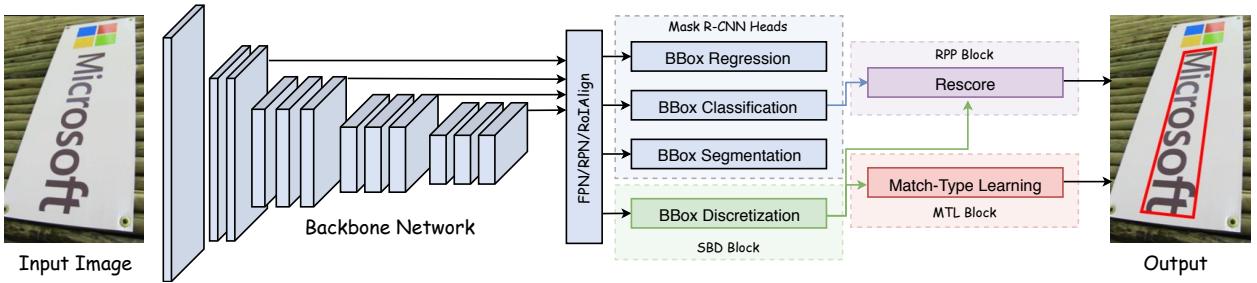
Unlike these methods, our method is the first to directly produce a compact quadrilateral bounding box without complex post-processing. Moreover, it can completely avoid inconsistent labeling issues.

3 Our Method

Our proposed scene text detection system consists of three core components: an Orderless Box Discretization (OBD) block, a matching-type learning (MTL) block, and re-scoring and post-processing (RPP) block. Figure 3 illustrates the overall pipeline of the proposed framework, and more details are presented in the following sections.
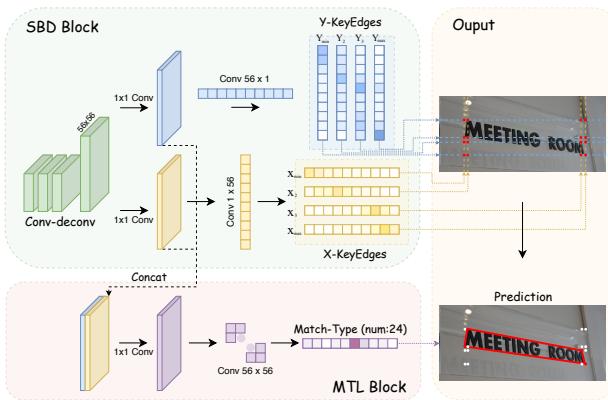
3.1 Orderless Box Discretization

The purpose of multi-orientation scene text detection is to accurately localize the textual content by generating outputs in the form of rectangular or quadrilateral bounding boxes. Compared with rectangular annotations, quadrilateral labels demonstrate an increased capability to cover effective text regions, especially for rotated texts. However, as discussed in Section 2, simply replacing rectangular bounding boxes with quadrilateral annotations can introduce inconsistency because of the sensitivity of the non-segmentation-based methods to label sequences. As shown in Figure 1, the detection model might fail to obtain accurate features for the corresponding points when facing small disturbances. One possible reason behind this is that the neural-network-based regressor for bounding box prediction is essentially a nonlinear continuous function, which means that each input is only mapped to one output. Thus a non-function or a function with a steep gradient cannot be effectively fitted. In our case, a small disturbance may completely change the whole sequence of the vertex and thus a similar input may result in completely different output as well as a steep gradient. Therefore, instead of predicting sequence-sensitive distances or coordinates, an OBD block is proposed to discretize the quadrilateral box into eight *Key Edges* (KE) comprising order-irrelevant points; *i.e.*, minimum $x(x_{min})$ and $y(y_{min})$), the second-smallest $x(x_2)$ and $y(y_2)$, the second-largest $x(x_3)$ and $y(y_3)$, and the maximum $x(x_{max})$ and $y(y_{max})$ (see Figure 1). We use x-KEs and y-KEs in the following sections to represent $[x_{min}, x_2, x_3, x_{max}]$ and $[y_{min}, y_2, y_3, y_{max}]$, respectively.

Fig. 3: Overview of the proposed detection framework.

Specifically, the proposed approach is based on the widely used generic object detection framework, Mask R-CNN [20]. As shown in Figure 4, the proposals processed by RoIAlign are fed into the OBD block with the pooling size of 14×14, where the feature maps are forwarded through four convolutional layers with 256 output channels. The output features are then upsampled by a $2\times$ deconvolutional layer and a $2\times$ bilinear upscaling layers. Thus, the output size of the feature maps F_{out} is $M \times M$, where M is 56 in our implementation. Furthermore, two convolution kernels shaped as $1 \times M$ and $M \times 1$ with six channels are employed to shrink the horizontal and vertical features for the x-KEs and y-KEs, respectively. Finally, the OBD model is trained by minimizing the cross-entropy loss L_{ke} over an M-way softmax output, where the corresponding positions of the ground-truth KEs are assigned to each output channel.

In practice, OBD does not directly learn the x-KEs and y-KEs because of the restriction of the region of interest (RoI). Specifically, the original Mask R-CNN framework limits the prediction inside the RoI areas. Thus, if the regression bounding box is not accurate, the missing pixels outside of the bounding box will not to be restored. To solve this problem, the x-KEs and y-KEs are encoded in the form of "half lines" during training. Suppose we have x-KEs, $x^i \in [x_{min}, x_2, x_3, x_{max}]$, and y-KEs, $y^i \in [y_{min}, y_2, y_3, y_{max}]$.

Fig. 4: Illustration of the OBD and MTL blocks.

Then, the "half lines" are defined as follows:

$$x_{half}^i = \frac{x^i + x_{mean}}{2},$$

$$y_{half}^i = \frac{y^i + y_{mean}}{2}, \quad (1)$$

where x_{mean} and y_{mean} represent the value of the mean central point of the ground-truth bounding box for the x and y axes, respectively. By employing such a training strategy, the proposed OBD block can break the RoI restriction (see Figure 5). Thus, it is more likely to produce accurate bounding box because x_{half} and y_{half} fall into the area of the RoIs in most cases, even if the border of the text instance is located outside the RoIs.

Similar to Mask R-CNN, the overall detector is trained in a multi-task manner. Thus, the loss function comprises four terms:

$$L = L_{cls} + L_{box} + L_{mask} + L_{ke}, \quad (2)$$

where the first three terms, L_{cls}, L_{box} and L_{mask}, follow the same settings as presented in [20]. L_{ke} is the cross-entropy loss, which is used for learning the Key Edges prediction task. The authors made an interesting observation in which the additional keypoint branch can harm the bounding box detection performance [20]. However, based on our experiments (see Tables 1 and 2), the proposed OBD block is the key component that significantly boosts the detection accuracy. There may be two reasons for this. First, ours is different from the keypoint detection task, which has to learn M^2 classes against each other. Thus, the numbers of competitive pixels in the OBD block is only M. Second, for the keypoint detection task, neither one-hot point nor a small circled area can be used to describe the target keypoint accurately, while the KEs produced by OBD are well defined. Thus, our method may provide more accurate supervision for training the network.
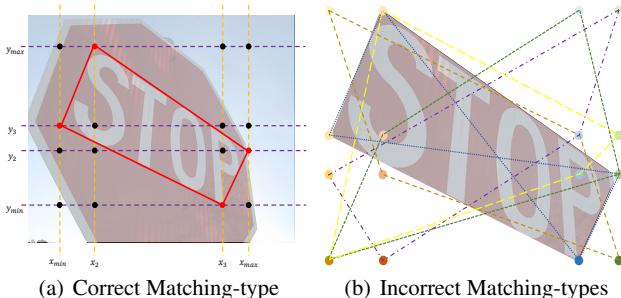
3.2 Matching-Type Learning

It is noteworthy that the OBD block only learns to predict the numerical values of eight KEs but is unable to predict
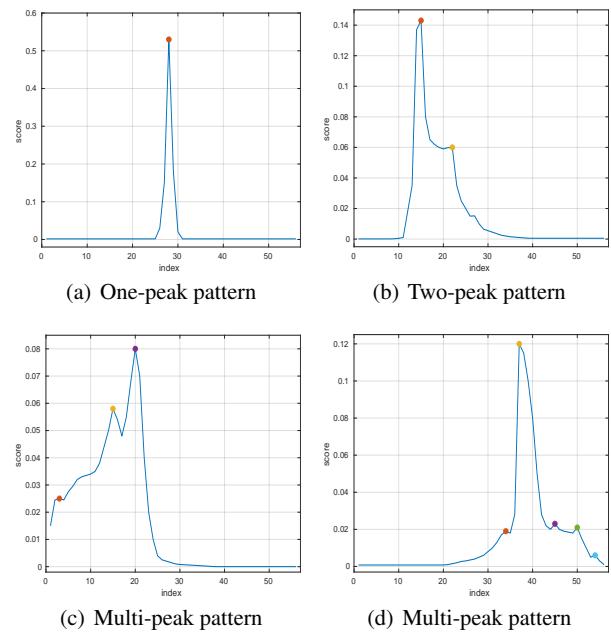
Fig. 5: The proposed framework can break the restrictions of the RoIs. The green solid quadrilateral and red dashed rectangular boxes represent the predictions and proposals, respectively.

(a) One-peak pattern   (b) Two-peak pattern

(c) Multi-peak pattern   (d) Multi-peak pattern

Fig. 7: Different patterns of S_{OBD} outputted by OBD block. (a) is the normal case while (b)(c)(d) are abnormal cases.

the connection between the x-KEs and y-KEs. Therefore, we need to design a proper matching procedure to reconstruct the quadrilateral bounding box from the KEs. Otherwise, the incorrect matching type may lead to completely unreasonable results (see Figure 6).

As described in Section 3.1, there are four x-KEs and four y-KEs outputted by the OBD block. Each x-KE should match one of the y-KEs to construct a corner point, such as (x_{min}, y_{min}), (x_2, y_{max}), and (x_{max}, y_2). Then, all four constructed corner points are assembled for the final prediction, giving us the quadrilateral bounding box. It is important to note that different orders of the corners would produce different results. Hence, the total number of matching-types between the x-KEs and y-KEs can be simply calculated by $A_4^4 = 24$. For example, the predicted matching-type in Figure 6(a) is $[(x_{min}, y_2), (x_2, y_{max}), (x_3, y_{min}), (x_{max}, y_3)]$. Based on this, a simple yet effective MTL module is proposed to learn the connections between x-KEs and y-KEs. Specifically, as shown in Figure 4, the feature maps that are used for predicting the x-KEs and y-KEs are used for classifying the matching-types. Specifically, the output feature of the deconvolution layer is connected to a convolutional layer having an $M/2 \times M/2$ kernel size with 24 output channels. Thus, the matching procedure is formed as a 24-category classification task. In our method, the MTL head is trained by minimizing the cross-entropy loss, and the experiments demonstrate that the convergence speed is very fast.

3.3 Re-scoring and Post-processing

The fact that the detectors can sometimes output high confidence scores for false positive samples is a long-standing issue in the detection community for both generic objects and text. One possible reason for this may be that the scoring head used in most of the current literature is supervised by the softmax loss, which is designed for classification but not for explicit localization. Moreover, the classification score only considers whether the instance is foreground or background, and it shows less sensitivity to the compactness of the bounding box.

Therefore, a confidence RPP block, is proposed to suppress unreasonable false positives. Specifically, RPP adopts a policy similar to multiple expert systems to reduce the risk of outputting high scores for negative samples. In RPP, an OBD score S_{OBD} is first calculated based on eight KEs (four x-KEs and four y-KEs):

$$S_{OBD} = \frac{1}{K} \sum_{k=1}^{K} \max_{v^k} f\left(v^k\right), \tag{3}$$

where $K = 8$ is the number of KEs, v^k is the output score vector of the kth KE shown in (4), and $f(v^k)$ is defined to sum up the peak value, v_i, and its neighbors. As shown in Figure 7(a), the distribution of S_{OBD} demonstrates a one-peak pattern in most cases. Nonetheless, the peak value is still significantly lower than 1. Hence, we sum up four adjacent scores that are near the peak value for each KE score to

(a) Correct Matching-type   (b) Incorrect Matching-types

Fig. 6: Illustration of different matching types.

avoid a confidence score that is too low.

$$v^k = [v_1, v_2, ..., \underbrace{v_{i-2}, v_{i-1}, v_i, v_{i+1}, v_{i+2}}_{f(v^k) = \Sigma_{p=max(i-2,1)}^{P=min(n,i+2)}(v_p)}, ...v_n]. \tag{4}$$

It is important to note that the number of adjacent values will be less than four if the peak value is located at the head or tail of the vector. Thus, only the existing neighbors should be counted. Finally, the refined confidence can be obtained by:

$$score = \frac{(2-\gamma)S_{box} + \gamma S_{OBD}}{2}, \tag{5}$$

where $0 \le \gamma \le 2$ is the weighting coefficient and S_{box} is the original softmax confidence for the bounding box. Because both S_{box} and S_{OBD} are both between [0,1], the value of $score(\Re)$ is also between [0,1]. Counting the S_{OBD} into the final score enables the proposed detector to draw lessons from multiple agents (eight KE scores) while enjoying the benefits of a tightness-aware confidence supervised by the KE prediction task.

3.4 Discussion

It has been proven that the detection performance can be often boosted with the multi-task learning framework. For example, as shown in [20], simultaneously training a detection head with an instance segmentation head can significantly improve the detection accuracy. Similarly, a segmentation head is also employed in the proposed OBD network to predict the area inside the bounding box, which forces the model to regularize pixel-level features to enhance both performance and robustness. However, some issues associated with the segmentation head are highlighted in Figure 8. In (a), the segmentation mask can sometimes produce false positive pixels while the OBD prediction remains correct. In (b), the segmentation head fails to maintain some positive samples that have been successfully detected by the OBD block. Therefore, compared with some segmentation-based approaches that directly reconstruct the bounding box by exploiting the segmentation mask, the MTL block can learn geometric constraints to avoid false positives caused by an inaccurate segmentation output. This also reduces the heavy reliance on the segmentation task. Specifically, as shown in Figure 6(b), the blue dashed line matches an invalid shape that violates the definition of a quadrilateral, because the sides should only have two intersections, at the head and tail. By simply removing these abnormal results, the MTL block can further eliminate some false positives that might cheat the segmentation branch.

Another interesting observation is that the RPP block exhibits a strong capability to suppress false positives, making

predictions more reliable. To provide an analysis, we visualize the term S_{OBD}, which is used in the RPP block (see Equation (5)). Doing so, we find that there are two typical patterns for the KE scores output by the OBD block, as shown in Figure 7. Sub-figure (a) shows a one-peak pattern, and sub-figure (b) shows a multi-peak pattern. In normal cases, the KE scores show a regular pattern, in which there is only one peak value in the output vector (see Figure 7(a)). However, with hard negative samples, two or more peak values appear (see Figures 7(b), 7(c), and 7(d)). These multiple peaks share confidence, and the total score is normalized to one. Therefore, based on Equations (3) and (5), the final score will be decreased such that the proposed model is less likely to output high confidence for those false-positive instances.

Based on our observation, we find that the matching-type prediction could be wrong even if KE is accurate. An example is shown in the bottom instance of the lower-right corner image of Figure 14 (b), where x_{min} is mistakenly matched to y_{min}. If x_{min} and the second smallest x change their matching y key edge, the detection result can be tighter. Although such a case does not obviously affect both the detection and recognition performance, it is an underlying weakness of the MTL. It is worth mentioning that sometimes the matching type may form an irregular bounding box, *i.e.*, the sides have self-intersection. We find that such cases are very rare and mostly occur with false negatives. For such irregular results, we simply remove them.

4 Ablation studies

4.1 Implementation details

Our model is implemented using PyTorch. We first evaluate the proposed components of our methods. The initial learning rate is set to 0.01, which is decreased by 10 at 10,000 iterations and 15,000 iterations. The maximum iterations is 20,000 and the image batch size is set to 4. The shorter size of the input image is randomly scaled from 680 to 1000 with the interval of 40, while the maximum size is set to 1480. The weights of KE and matching type learning are set to 0.1 and 0.01, respectively. Flip, random crop, and random rotation are used to improve the generalization ability. Unless specified otherwise, the re-scoring ratio is kept to be 1.4.

For ablation studies of refinements, each experiment uses a single network that is a variation of our baseline model (first row of Table 5). Each network is trained on the official ReCTS training set unless specified otherwise. Additionally, because the test scale may significantly influence the final detection result, the testing max size is fixed at 2,000 pixels, and the scale is fixed to 1,400 pixels for strictly fair ablation experiments. The ratio of the flip is also fixed at 0.5, which is the flipping probability for deciding whether

Fig. 8: Compared with the segmentation head, the proposed KE head predicts more compact bounding boxes and shows a higher recall rate for instances that were missed by segmentation. Colored quadrangles are the final detection results, whereas white transparent areas are the mask predictions grouped by the minimum area rectangle.

to horizontally flip the images for data augmentation. Results are reported on the validation set of ReCTS based on the widely used main performance metric, Hmean. We also report the best confidence threshold that leads to the best performance, which can also reveal some important information.

The number of iterations for training one network is set to 80,000 iterations, with a batch size of four images per GPU on four 1080ti GPUs. The final cumulative model is trained for 160 epochs on four V100 GPUs, which takes approximately 6 days. The baseline model employs ResNet-101-FPN as the backbone, which is initialized by a model pretrained on the MLT [27] data. We only use fixed batch normalization for the stem and bottleneck, i.e., the batch statistics and the affine parameters are fixed. For all prediction heads, we do not use batch normalization.

4.2 Ablation studies of the proposed method

In this section, we report ablation studies on the ICDAR 2015 [26] dataset, to validate the effectiveness of each component of our method. First, we evaluate the influence of the proposed modules on performance. The results are presented in Table 1 and Figure 9. From Table 1, we can see that OBD and RPP can lead to improvements of 2.4 and 0.6%, respectively, in terms of Hmean. Additionally, figure 9 shows that our method can substantially outperform the baseline Mask R-CNN under different confidence thresholds, further demonstrating its effectiveness.

Furthermore, we conduct experiments by comparing the mask and KE branches (including OBD and RPP) on the same network. Thus, we test only on one of the branches. We simply use the provided training samples of IC15 without any data augmentation. The results are presented in Table 2, verifying that the proposed modules can effectively improve the scene text detection performance.

More importantly, we also conduct experiments to verify that introducing ambiguity in the training is harmful to

Table 1: Ablation studies demonstrating the effectiveness of the proposed method. The γ of RPP is set to 1.4 (best practice). The results on this table also adopt MLT training data and data augmentation strategies to help improve the final performance.

Datasets	Algorithms	Hmean
ICDAR2015	Mask R-CNN baseline	83.5%
	Baseline + OBD	85.9% (↑ **2.4%**)
	Baseline + OBD + RPP	86.5% (↑ **3.0%**)

Table 2: Ablation studies for comparing the mask branch and KE branch. The γ of RPP is set to 0.8 (best practice). Compared to Table 1, the results here are all tested in different branches of the same model without any data augmentation.

Datasets	Algorithms	Hmean
ICDAR2015	Mask branch	79.4%
	KE branch without RPP	80.4% (↑ **1.0%**)
	KE branch with RPP	81.0% (↑ **1.6%**)

Table 3: Comparison on ICDAR 2015 dataset showing different methods' ability of resistant to the inconsistent labeling issue (by adding rotated pseudo samples). TB: Textboxes++. LD: using lower rotation degrees.
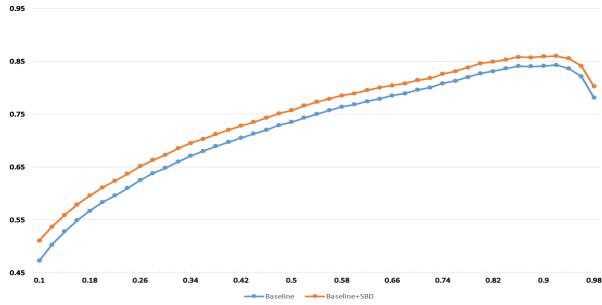
	TB	East	CTD	APE	Ours
Hmean (baseline)	80.1%	78.3%	74.7%	79.4	80.4%
Hmean (rotation)	70.4%	64.6%	50.1%	77.4	80.7%
Variance	↓ **9.7%**	↓ **13.7%**	↓ **24.6%**	↓ **2.0%**	↑ **0.3%**
Hmean (LD)	79.5%	76.0%	68.5%	80.1%	81.5%
Variance (LD)	↓ **0.6%**	↓ **2.3%**	↓ **6.2%**	↑ **0.7%**	↑ **1.1%**

achieving good results. Specifically, by using the same configuration, we first train Textboxes++ [14], EAST [25], CTD [29], APE [39] (the champion method of DOAI2019 competition task1), and the proposed method with the original 1,000 training images of the ICDAR 2015 dataset. Then, we randomly rotate the training images $[0°, 15°, 30°, ..., 360°]$ and randomly select additional 2,000 images from the rotated dataset to fine-tune these models. We also randomly select additional 2,000 images that are between $[-30°, 30°]$ to evaluate the difference under lower rotation degree. The results are presented in Table 3. Our method can effectively address the inconsistent labeling issue without drastically degrading the accuracy. Furthermore, as shown in Table 4, our proposed method exhibit higher robustness under various degrees of rotation.

Note for the resnet-50 version and the following final competition version of our method, the inference time is 4.5 FPS and 0.83 FPS, respectively. The speed is tested using a single NVIDIA GTX 2080 Ti and the short size of the input image is scaled to 1,000.

Table 4: Hmean results under different rotation degrees on ICDAR 2015 dataset. The rotation angle represents the value used for the data augmentation during the training phase.

	5°	30°	60°	90°
Ours	↑0.9%	↑1.1%	↑1.3%	↑0.3%

Fig. 9: Ablation study on the ICDAR 2015 benchmark. X-axis represents confidence threshold and Y-axis represents Hmean result. Baseline represents Mask R-CNN. By integrating with proposed OBD, the detection results can be substantially better than the results of the Mask R-CNN baseline.

Fig. 10: Example images of the ReCTS. Small, stacked multi-orientation, illumination, and annotation ambiguity are the main challenges for this dataset.

4.3 Ablation studies of refinements based on our method

In this section, we provide a detailed analysis of the impact of refinements based on the proposed methods, to evaluate the limits of our method and whether it can be mutually promoted by existing modules. By combining effective refinements, our method achieves first place in the detection task of the ICDAR 2019 Robust Reading Challenge on Reading Chinese Text on Signboards.

In the following sections, we present an extensive set of experiments that rate our baseline model. Thus, we present results of OBD having alternative architectures and different strategies with respect to six relevant components for training, including data arrangement, pre-processing, backbone, proposal generation, prediction head, and post-processing.

The objective is to show that the proposed model corresponds to a local optimum in the space of architectures and parameters and to evaluate the sensitivity of the final performance to each design choice. The following discussions follow the structure of Table 5. Note that the significant breadth and exhaustivity of the following experiments represent more than 3,000 GPU hours of training time.

4.3.1 Competition Dataset

The competition dataset, Reading Chinese Text on Signboards (ReCTS), is a practical and challenging multi-orientation natural scene text dataset containing 25,000 signboard images. A total of 20,000 images are used for the training set, with a total of 166,952 text instances. The remaining 5,000 images are used for the test set. Examples of this dataset are shown in Figure 10. The layout and arrangement of Chinese characters in this dataset are clearly differ-

ent from those in other benchmarks. Because the function of a signboard is to attract a customer base, it is very common to notice their aesthetic appearance. Thus, the Chinese characters can be arranged in any kind of layout with various fonts. Additionally, characters from one word can be in diverse orientations, diverse fonts, or diverse shapes, which complicates the challenge. This dataset provides both text lines and character annotations to inspire new algorithms that can take advantage of the arrangement of characters. To evaluate the function of each component, we split the original training set into 18,000 training images and 2,000 validation images.

4.4 Ablation study of data arrangement

Considering the image diversity and the consistency and quality of annotation, we collected a 60,000-item dataset for pretraining, which consisted of 30,000 images from the LSVT [30] training set, 10,000 images from the MLT 2019 [32] training set, and 5,603 images from ArT [31], which contained all the images from SCUT-CTW1500 [29] and Total-text [28, 40]. The remaining 14,859 images were selected from RCTW-17 [41], ICDAR 2015 [26], ICDAR 2013 [42], MSRA-TD500 [43], COCO-Text [44], and USTB-SV1K [45]. Note that we transferred polygonal annotations to the minimum area rectangle for training.

The ablation results are presented in Table 5. If we only were to use the pretrained data without the split training data from the ReCTS, the result in the ReCTS validation set would be significantly worse than that of the baseline, even if the pretrained model were trained with more iterations. This is because the diversity and annotation granularity of the selected pretrained dataset is still very different from that of the ReCTS dataset. However, using the

Table 5: Ablation studies of different refinements based on our method. Each variation is evaluated on the ReCTS validation set. It is worth mentioning that we regard difficult samples as true negatives in the validation because they cannot be recognized and only loosely annotated in the competition dataset. However, in the final ranking, detection box in the difficult region are set to "do not care", which can result in a leap improvement. We evaluate variations of our baseline model (second row). Every row corresponds to one variation in different part. We train each variation with ResNet-101-FPN and fixed random seeds and equal 80,000 iterations (unless specifying) and report Hmean in the best confident threshold (grid search).

Methods	Best threshold	Recall (%)	Precision (%)	Hmean (%)	ΔHmean
Baseline model (based on OBD [12])					
with mlt pretrained model					
with flip (0.5)	**0.91**	**78.1**	**80.1**	**79.1**	**-**
test scale: min size: 1400; max size: 2000.					
Data arrangement					
With data cleaning	0.93	77.7	80.3	79.0	↓ **0.1**
With only mlt pretrained data (100k iters)	0.97	53.4	56.1	54.7	↓ **24.4**
With only 60k pretrained data (200k iters)	0.81	50.8	61.0	55.5	↓ **23.6**
With defect data	0.91	75.8	72.5	74.1	↓ **5.0**
Without MLT data pretrain	0.85	75.5	81.9	78.6	↓ **0.5**
With 60k pretrained model	0.91	78.8	81.9	80.3	↑ **1.2**
Pre-processing					
With random crop (best ratio)	0.91	78.4	83.7	81.0	↑ **1.9**
With random rotate (best ratio)	0.91	77.6	81.8	79.7	↑ **0.6**
With color jittering	0.91	76.4	82.5	79.3	↑ **0.2**
With medium random scale training					
ori: (560,600,...,920,) max: 1300	0.89	80.3	82.2	81.3	↑ **2.2**
to: (680,720,...,1120,) max: 1800					
With large random scale training					
ori: (560,600,...,920,) max: 1300	0.89	80.2	83.6	81.9	↑ **2.8**
to: (800,840,...,1400,) max: 2560					
Backbone					
With ResNext-152-32x8d-FPN-IN5k					
(using detectron pretrained model) v1	0.91	79.4	84.0	81.6	↑ **2.5**
With ASPP in KE head	0.91	76.1	80.1	78.0	↓ **1.1**
With ASPP in (backbone 1/16)	0.89	73.1	81.3	77.0	↓ **2.1**
With deformable convolution (C4-1)	0.87	79.5	83.9	81.7	↑ **2.6**
With deformable convolution (C4-2)	0.89	79.1	84.3	81.6	↑ **2.5**
With deformable convolution (C3-)	0.83	81.2	81.9	81.6	↑ **2.5**
With panoptic segmentation (dice loss)	0.67	77.7	80.3	79.0	↓ **0.1**
With pyramid attention network (PAN)	0.85	77.6	83.1	80.3	↑ **1.2**
With multi-scale network (MSN)	0.91	79.0	81.6	80.3	↑ **1.2**
Proposal generation					
With deformable PSROI pooling	0.91	80.7	79.4	80.0	↑ **0.9**
Prediction head					
With character head	0.93	77.7	82.0	79.8	↑ **0.7**
With OHEMv1	0.59	76.9	80.0	78.4	↓ **0.7**
With OHEMv2	0.65	75.8	81.1	78.3	↓ **0.8**
With OHEMv3	0.55	77.5	79.8	78.6	↓ **0.5**
With mask scoring	0.93	75.7	81.8	78.6	↓ **0.5**
With cascade r-cnn (ensemble)	-	77.7	80.3	79.0	↓ **0.1**
Post-processing					
With polygonal non-maximum suppression	0.91	77.2	82.8	79.9	↑ **0.8**
With Key Edge RPP	0.91	78.5	79.9	79.2	↑ **0.1**
Final model					
accumulating effective modules	0.91	83.2	89.5	86.2	↑ **7.1**

model trained with pretrained data is better than using the ImageNet model. For example, when directly using the ImageNet ResNet-101 model instead of the MLT pretrained model from the baseline method, the Hmean is reduced by 0.5%. Using the model having 60,000 pretrained data, followed by finetuning the model on the split ReCTS training data improved the result by 1.2% in terms of Hmean. To evaluate the importance of the data quality, we mimicked the manual annotation error by removing 5% of the training annotation instances and did not correct some samples with annotation ambiguity from the original ReCTS training data. The results indicate that using defective training data significantly degrades the performance.
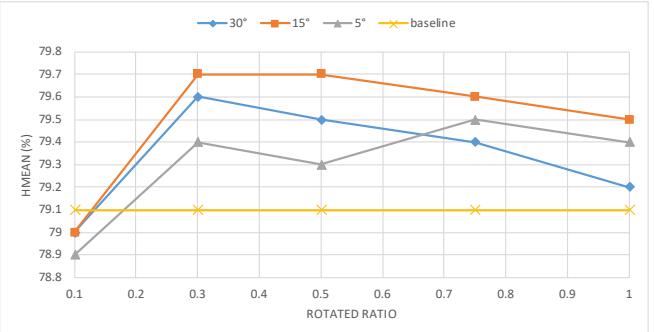
4.5 Ablation study of pre-processing

Our baseline model used a pretrained model having only a flip strategy for data augmentation. We compared the baseline with various other data augmentation methods.
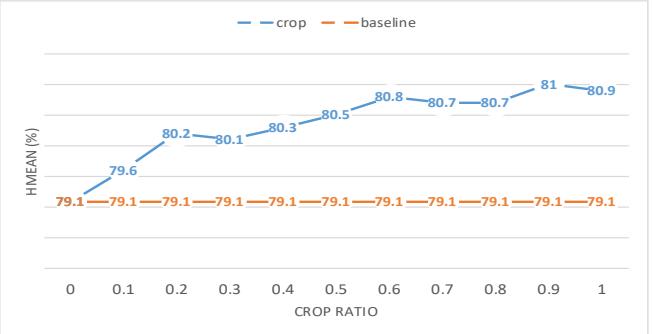
Cropping and rotation. Without introducing extra parameters or training/testing times, the results presented in Table 5 verify that both rotation and data cropping augmentation strategies improved the detection results. We further conducted a sensitivity analysis of how the ratios of using these two strategies influence the performance, as shown in Figure 11. Some useful findings can be derived from Figure 11(a), as summarized below.

- With appropriate ratios, three rotated degrees (30°, 15°, and 5°) outperformed the baseline method in most ratios, with 0.5, 0.6, and 0.4%, respectively.
- Under a 0.1 rotated ratio, the performances with the three rotated degrees were all worse than the baseline. This may be because the pseudo samples changed the distribution of the original dataset, whereas very few pseudo samples were insufficient to improve the generalization ability. Conversely, the ratios to achieve the best results for various rotated degrees always lie between 0.3 and 0.8, which empirically suggests that using a medium ratio for the rotated data augmentation strategy might be a suitable choice.
- We can also see that the performance using a rotated angle of 15° was consistently better than that with 30° and 5°.

Compared with the rotated data augmentation strategy, the random cropping strategy significantly improved detection performance. The best performance, as shown in Table 5, achieved a 1.9% improvement in terms of Hmean, compared with the baseline method. Sensitivity analysis, as shown in Figure 11(b), was also conducted, revealing that, as

(a) Ablation results of rotated pseudo samples.

(b) Ablation results of crop pseudo samples.

Fig. 11: Ablation studies of data augmentation strategies.

the crop ratio improved, the performance also tended to improve. The result suggests that always using the crop strategy was conducive to improving the detection results. Note that a crop ratio of 0.1 only improved the Hmean by 0.5%, whereas other ratios improved it by more than 1%, which is similar to the phenomenon when using a rotated ratio 0.1.

Color jittering. We also conduct a simple ablation study to evaluate the performance of color jittering. Based on the same settings as of the baseline method, we empirically set the ratios of brightness, contrast, saturation, and hue to 0.5, 0.5, 0.5, and 0.1, respectively. The ratio represents the degree of disturbance of each specific transformation. The results in Table 5 indicate that using color jittering data augmentation slightly improved the result by 0.2% in terms of Hmean.

Training image scale. The training image scale/size is specifically important for a scene text detection. To evaluate how the training scale influences the results of our method, we used two parameters (*i.e.*, *scale* and *MaxSize*) to control the training scale. The first item resized the minimum side of the image to a specific parameter.

In our implementation, there are a set of values for random scaling. The second item restricts the maximum size of the image sides. The value of *scale* must be less than *MaxSize*, and the entire scaling process strictly retains the

Table 6: Ablation experiments for large scale training. Hmean$_1$, Hmean$_2$, and Hmean$_3$ represent default training scale, medium training scale, and large training scale, respectively. The first row compares the performance based on the baseline setting. The other three rows are the best setting (using grid search to find the best *scale* and *MaxSize*) for each training scale.

(*Scale*, *MaxSize*)	Hmean$_1$ (%)	Hmean$_2$ (%)	Hmean$_3$
(1400, 2000)	79.1	81.3	81.9
(800, 1300)	81.5	-	-
(1600, 1600)	-	82.2	-
(1600, 1700)	-	-	82.5

original aspect ratio. We primarily compare three different settings: the default training scale (*scale*: 560 to 920 with intervals of 40, *MaxSize* was 1,300); medium training scale (*scale*: 680 to 1,120 with intervals of 40, *MaxSize* was 1,800); and large training scale (*scale* 800 to 1,400 with intervals of 40, *MaxSize* was 2,560).

The results are presented in Table 6, which verify the following: 1) a larger training scale requires a larger testing scale for the best performance. 2) As the larger training scale increases, so does the performance. Note that, although a larger training scale can improve performance, it is costly and may require significantly more GPU memory.

4.6 Ablation study of the backbone

A well-known hypothesis is that a deeper and wider network architecture delivers better performance than does a shallower and thinner one. However, increasing the network depth naively will significantly increase the computational cost with only limited improvement. Therefore, we investigate different types of backbone architectures. The results are shown in Table 5 and are summarized as follows:

– By changing the backbone, ResNet-101-FPN of the baseline model into a ResNeXt-152-32x8d-FPN-IN5k, Hmean can be increased by 2.5%. Note that the pretrained model of ResNeXt-152-32x8d-FPN-IN5k was pretrained on ImageNet using the Facebook Detectron framework.

– Atrous spatial pyramid pooling (ASPP) [46] is effective in semantic segmentation, which is known for its function in increasing the receptive field. However, in this scene text detection task, using ASPP in the KE head or backbone reduced performance by 1.1 and 2.1%, respectively. One possible reason is that the change in network architecture usually requires more iterations. However, the best confidence thresholds for the best performance using ASPP were 0.91 and 0.89, which are similar to the best threshold of the baseline model, suggesting that the network had already converged.

– Deformable convolution [47] is an effective module used for many tasks. It adds 2D offsets to the regular sampling grid of the standard convolution, allowing free form deformation of the convolutional operation. This is suitable for scene text detection, owing to the mutable characteristics of the text. We experimented with three methods of deformable convolution by adding deformable convolutions from the C4-1, C4-2, and C3 of the backbone, and the results show that the performance could be significantly improved by 2.6, 2.5, and 2.5%, respectively, in terms of Hmean.

– Motivated by the panoptic feature pyramid networks [48], we also tested whether a panoptic segmentation loss was useful for scene text detection. To this end, we used a dice loss in the output of the FPN for panoptic segmentation, which had two classes: background and text. The result in Table 5 indicates that Hmean was reduced by 0.1%. However, the best threshold was 0.67, which indicates that the background noise may have somehow reduced the confidence of the training procedure.

– The pyramid attention network (PAN) [49] is a novel structure that combines an attention mechanism and a spatial pyramid to extract precise dense features for semantic segmentation tasks. Because it can effectively suppress false alarms caused by text-like backgrounds, we integrated it into the backbone and tested its function. The results show that using PAN led to a 1.2% improvement in terms of Hmean, but it also increased the computational cost with an increase of 2.4 GB video memory.

– The multi-scale network (MSN) [22] is robust for scene text detection because it employs multiple network channels to extract and fuse features at different scales concurrently. In our experiment, integrating MSN into the backbone also increased the performance by 1.2% in terms of Hmean. Note that, compared with PAN, the recall of the MSN was much better under a higher best threshold, which suggests that different architectures may have had different functions related to the performance of the detector.

4.7 Ablation study on proposal generation

The proposed model is based on a two-stage framework, and the region proposal network (RPN) [50] is used as the default proposal generation mechanism.

Previous studies have modified the anchor generation mechanism, including DMPNet [8], DeRPN [51], Kmeans anchor [52], scale-adaptive anchor [53], and guided anchor [54], to improve the results. For simplicity, we retrain the default RPN structure with the statistical setting of the anchor box based on the training set.

Table 7: Ablation results of using cascade r-cnn. cf: best threshold. R: recall. P: precision. H: Hmean.

Method	cf	R (%)	P (%)	H (%)	ΔH
Baseline model	0.91	78.1	80.1	79.1	-
Stage 1	0.91	74.7	81.8	78.1	↓ **1.0**
Stage 2	0.87	76.3	81.1	78.6	↓ **0.5**
Stage 3	0.87	75.9	79.5	77.7	↓ **1.4**
ensemble	-	77.7	80.3	79.0	↓ **0.1**

Table 8: Ablation experiments for using character head. H: Hmean.

Method	H (%)	ΔH
Baseline	79.1	-
Baseline + character head	79.8	↑ **0.7**
Baseline + character head + mask character	79.8	↑ **0.7**
Baseline + character head + instance connection	79.6	↑ **0.5**
Baseline + character head + instance connection - KE head	75.2	↓ **3.9**

The other important part in this proposal generation stage is the sampling process, (e.g., RoI pooling [50], RoI align [20] (our default setting), and PSRoI pooling [55]. We choose to evaluate Deformable PSRoI Pooling [47] for our method, because it has been effective for scene text detection [56], and the flexible process may be beneficial to the proposed OBD. The result is shown in Table 5: using deformable PSRoI Pooling improved the baseline method by 0.9% in terms of Hmean.

4.8 Ablation study on the prediction head

The final part of the two-stage detection framework is the prediction head. To clearly evaluate the effectiveness of the components, ablation experiments are separately conducted on different heads.

Box head. Empirically, online hard negative examples mining (OHEM) [57] is not always effective with respect to different benchmarks. For example, using the same framework minus the training data can significantly improve the results with the ICDAR 2015 benchmark [26] while reducing the results on the MLT benchmark [27]. This finding may be related to the data distribution, which is difficult to trace.

Thus, we test three versions of the OHEM in the validation set. The first version, OHEMv1, is the same as the original implementation; the second version, OHEMv2, simply ignores the top 5 hard examples to avoid outliers. These two versions have the same ratio, which is set to 0.25. The third version, OHEMv3, simply uses a higher ratio (0.5) to guarantee more hard samples and less easy samples. The results in Table 5 show that three versions all reduce Hmean, by 0.7, 0.8, and 0.5, respectively. Note that using OHEM will also result in the reduction of the best confidence, which means that the forced learning of hard examples can reduce the confidence of normal examples. Conversely, we also evaluated the performance of the cascade R-CNN, and the results are shown in Table 7. However, the results show that using a cascade does not result in further improvements.

Mask head. To improve the mask head, we evaluate two methods (*i.e.*, mask scoring [49]), as shown in Table 5. The results show that modification of the mask head does not
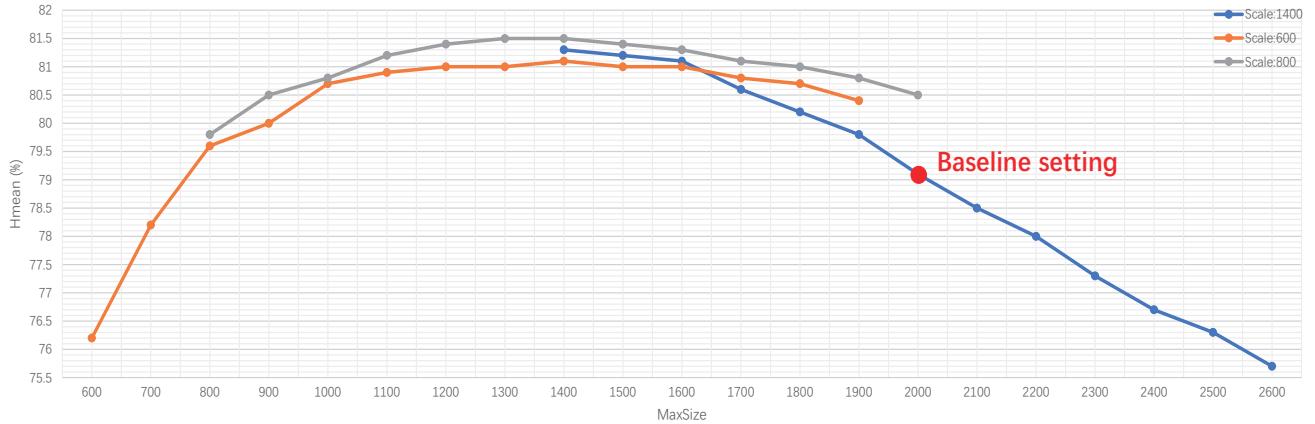
contribute to the detection performance. However, the mask prediction results are visually more compact and accurate compared with the baseline.

Character head. It is well known that stronger supervision can result in better performance. Because the competition also provides a character ground truth, we build and evaluate the performance of an auxiliary character head. The implementation of the character head is exactly the same as that for the box head, except for the ground truth. Unlike the box, mask, and KE head, the proposed character head is built on a different RPN. Thus, the character head does not share the same proposal with the other heads. The KE head directly produces a quadrilateral bounding box (word box) directly used for the final detection, and we test whether the auxiliary head could indirectly (shared backbone) improve the word-box detection performance.

The ablation results in Table 8 demonstrate this idea, which shows that using a character head improved the Hmean by 0.7%. Additionally, if we add a mask prediction head to the character head (*i.e.*, the mask character in Table 8), the result would remain the same. Moreover, we employ a triplet loss to learn the connection between the characters. The ground truth includes whether the characters belong to the same text instances. However, the improvement is decreased to 0.5%. This may be because the instance connection introduced an inconsistent labeling issue. We further test the performance using only the character head with an instance connection and without the KE head. Hmean is reduced by 3.9% compared with the baseline method, suggesting that using character as an auxiliary head instead of the final prediction head is a good choice.

4.9 Ablation study of post-processing

The last step is to apply post-processing methods for final improvement. To this end, we compare the baseline with a series of standard and more effective post-processing methods.

Fig. 12: Ablation study of the testing scale. Note that the training scale is the default setting mentioned in Section 4.5.

Polygonal non-maximize suppression (PNMS). Traditional non-maximum suppression (NMS) methods between horizontal rectangular bounding boxes can cause unnecessary suppression. Thus, we conduct ablation experiments to evaluate the performance of the PNMS. We use grid search to find the best threshold to find both NMS and PNMS for fair combination, which is 0.3 and 0.15, respectively. The result in Table 5 shows that using PNMS performs better than NMS by 0.8% in terms of Hmean. Additionally, PNMS is much more effective when using a test ensemble in practice.

Key edge RPP. The proposed key edge RPP proved effective on the ICDAR 2015 benchmark. Thus, we also test whether it applies to the competition dataset. The ablation result in Table 5 shows that it slightly improves the Hmean by 0.1% compared with the baseline. It is worth noticing that, although the best confidence threshold is 0.91, which is the same as that of the baseline, the recall is increased by 0.4% while only reducing the precision by 0.2%.

Large-scale testing. We also conduct experiments to evaluate how the testing scale influenced performance. The results are shown in Figure 12, which demonstrates that a proper setting of *scale* and *MaxSize* significantly improves the detection performance. Additionally, the results reveal that there is a limitation of the *MaxSize*. That is, if the value of *MaxSize* is higher than a certain value, the performance would be gradually reduced.

Test ensemble. To evaluate the performance of the test ensemble, we conduct ablation experiments with four different aspects: different backbone ensemble; multiple intermediate model ensemble; a multi-scale ensemble; and an independent model ensemble. Note that, to achieve the best performance, implementing ensemble or multi-scale testing requires some tricks. Otherwise, the results may be worse. We summarize the results as follows:

– Using a high confidence threshold. One weakness of multi-scale ensembling is that if a true-negative detection exists in one of the testing scales, it cannot be avoided unless we set a high confidence threshold to exclude it during the ensemble phase. Therefore, for each scale, we first test its best confidence threshold (cf) on the validation set. Then, we use a higher confidence for the model ensemble.

– Variant scale of multi-scale testing. The performance of small scale (600 (*scale*), 1200 (*MaxSize*)) is rated. For example, in the ReCTs competition, it is much worse than that of large-scale (1,600, 1,600). However, small scales are better for detecting large instances compared with large scales, and they can always be mutually promoted in practice.

– Using a strict PNMS threshold. A normal case for the ensemble result is that the recall can be significantly improved, whereas the prediction is dramatically reduced. When observing the final integrated detection boxes, it is easy to find that the reduction was caused by boxes-in-boxes and many stacked redundant boxes. Using a strict PNMS can effectively solve this issue.

Based on these principles, we conclude the results of the four ensemble aspects as follows.

– **Different backbone ensembles.** We train three models using the baseline setting with three types of deformable convolution, starting from C4-1, C4-2, and C3 of ResNet-101, respectively. The ensemble results of the three methods are shown in Table 9. From the table, we can see that integrating the models with a series of simple backbone modifications improved the detection performance, even based on a relatively high baseline. Additionally, the results show that integrating more components resulted in better performance.

– **Multiple intermediate model ensembles.** We also evaluate the performance of integrating intermediate models.

Table 9: Ablation experiments for different approaches of model ensemble. 'def': deformable convolution.

Method	Backbone ensemble			Intermediate model ensemble			Multi-scale ensemble (*scale, MaxSize*)			Model ensemble	
Components	def-C4-1	def-C4-2	def-C3	x152-60k	x152-70k	x152-80k	(600,1600)	(1200,1600)	(1600,1600)	M1	M2
Hmean (%)	81.7	81.6	81.6	80.7	81.7	81.6	79.8	82.1	82.6	83.2	83.5
Ensemble	def-C4-1 & def-C4-2	def-C4-1 & defC3	def-C4-1 & def-C4-2 & def-C3	x152-60k & x152-70k & x152-80k			(600, 1600) & (1200, 1600) & (1600, 1600)			M1 & M2	
Hmean (%)	81.8	82.1	82.2	81.9			83.2			83.7	

Table 10: Experimental results for the ICDAR 2015 dataset. R: recall. P: precision.

Algorithms	R(%)	P(%)	Hmean(%)
Tian et al. [6]	52.0	74.0	61.0
Shi et al. [9]	76.8	73.1	75.0
Liu et al. [8]	68.2	73.2	70.6
Zhou et al. [25]	73.5	83.6	78.2
Ma et al. [23]	73.2	82.2	77.4
Hu et al. [58]	77.0	79.3	78.2
Liao et al. [15]	79.0	85.6	82.2
Deng et al. [34]	82.0	85.5	83.7
Ma et al. [23]	82.2	73.2	77.4
Lyu et al. [35]	79.7	89.5	84.3
He et al. [18]	80.0	82.0	81.0
Xu et al. [59]	80.5	84.3	82.4
Tang et al. [60]	80.3	83.7	82.0
Wang et al. [37]	84.5	86.9	85.7
Xie et al. [11]	85.8	88.7	87.2
Zhang et al. [61]	83.5	91.3	87.2
Liu et al. [16]	87.9	91.9	89.8
Baek et al. [62]	84.3	89.8	86.9
Huang et al. [63]	81.5	90.8	85.9
Zhong et al. [64]	80.1	87.8	83.8
He et al. [65]	86.0	87.0	87.0
Liu et al. [66]	87.6	86.6	87.1
Liao et al. [14]	78.5	87.8	82.9
Long et al. [67]	80.4	84.9	82.6
He et al. [68]	79.7	92.0	85.4
Lyu et al. [69]	81.0	91.6	86.0
He et al. [17]	73.0	80.0	77.0
Wang et al. [70]	79.6	83.2	81.4
Liao et al. [71]	87.3	86.6	87.0
Wang et al. [72]	81.9	84.0	82.9
Wang et al. [73]	86.0	89.2	87.6
Qin et al. [74]	88.0	91.7	89.8
Feng et al. [75]	83.8	92.5	87.9
Liu et al. [12]	83.8	89.4	86.5
Ours	88.2	92.1	**90.1**

Table 11: Experimental results for the MLT dataset. SS represents a single scale. R: recall. P: precision. Note that we only used a single scale for all experiments.

Algorithms	R(%)	P(%)	Hmean(%)
linkage-ER-Flow [27]	25.59	44.48	32.49
TH-DL [27]	34.78	67.75	45.97
SARI FDU RRPN v2 [23]	67.0	55.0	61.0
SARI FDU RRPN v1 [23]	55.5	71.17	62.37
Sensetime OCR [27]	69.0	67.75	45.97
SCUT_DLVClab1 [8]	62.3	80.28	64.96
AF-RNN [76]	66.0	75.0	70.0
Lyu et al. [35]	70.6	74.3	72.4
FOTS [16]	62.3	81.86	70.75
CRAFT [62]	68.2	80.6	73.9
Liu et al. [12]	70.1	83.6	76.3
Ours	76.44	82.75	**79.47**

then integrated with a PNMS threshold 0.02 higher than the original best threshold, which resulted in approximate optimum integrating results with 0.6% improvement in terms of Hmean, as shown in Table 9.

– **Independent model ensembles.** Finally, we test the performance of integrating the two final models. The first model contains the baseline setting plus deformable convolution, and the second model contains the baseline setting with the ResNext-152 backbone. We independently integrate each model using an intermediate model ensemble and a multi-scale ensemble. Then, we assemble the final results of the two models. As shown in Table 9, the detection result can still be improved.

5 Comparison with state-of-the-art methods

To further evaluate the effectiveness of the proposed method, we carry out experiments and compare our final model with other state-of-the-art methods on three scene text datasets: ICDAR 2015 [26], MLT [27], and ReCTS (See Section 4.3.1). We also conduct an experiment on one aerial dataset, HRSC2016 [77], to further demonstrate the generalization ability of our method.

Final model. The final model is designed by combine the effective modules evaluated in Table 5. Specifically, based on the baseline setting, we refine our model in all six aspects. During the data arrangement stage, we use 60,000 pretrained data items to train a pretrained model for 200,000

We use the trained model with the ResNext-152 backbone as a strong baseline and selected the last three intermediate iterating models with 10,000 iterations as intervals for the ensemble. The results shown in Table 9 also demonstrate that when using the model ensemble, the intermediate models could be mutually promoted.

– **Multi-scale ensemble.** To evaluate the performance of the multi-scale ensemble, we use grid searching to find the best PNMS threshold for three specified settings (*scale, MaxSize*), representing large, medium, and small text instances, respectively. Each detection result was

Table 12: Competition results on the ReCTS dataset. The results are from the competition website https://tinyurl.com/ReCTS2019. For the detection task, the ranking is based on Hmean. For End-to-End detection and recognition task, the ranking is based on 1-NED. NED: normalized edit distance.

Affiliation	Detection Result			End-to-End Result			
	Recall (%)	Precision (%)	Hmean (%)	Recall (%)	Precision (%)	Hmean (%)	1-NED (%)
Ours	93.97	92.76	**93.36**	93.97	92.76	93.36	**81.62**
Tian et al.	93.46	92.59	93.03	92.49	93.49	92.99	81.45
Liu et al.	93.41	91.62	92.50	-	-	-	-
Zhu et al.	93.51	89.15	91.27	92.36	91.87	92.12	79.38
Mei et al.	91.96	90.09	91.02	-	-	-	-
Li et al.	90.03	91.65	90.83	90.80	90.26	90.53	73.43
Zheng et al.	89.84	91.41	90.62	-	-	-	-
Zhou et al.	90.99	89.59	90.28	90.99	89.59	90.28	74.35
Zhang et al.	93.66	86.35	89.86	93.62	87.22	90.30	76.60
Zhao et al.	86.13	92.72	89.31	86.12	92.73	89.30	72.76
Xu et al.	-	-	-	91.54	90.28	90.91	71.89
Wang et al.	88.92	88.70	88.80	88.89	88.92	88.91	71.81
Baek et al.	85.33	89.38	87.31	75.89	78.44	77.14	41.68
Wang et al.	84.67	89.53	87.03	84.64	89.56	87.03	71.10
Wang et al.	-	-	-	69.49	89.52	78.24	50.36
Li et al.	82.27	88.49	85.27	-	-	-	-
Xu et al.	88.52	79.32	83.66	-	-	-	-
Lu et al.	85.18	79.66	82.33	-	-	-	-
Ma et al.	83.16	80.77	81.94	-	-	-	-
Tian et al.	96.17	69.20	80.48	-	-	-	-
Feng et al.	73.05	78.35	75.61	-	-	-	-
Luan et al.	70.35	80.19	74.95	-	-	-	-
Yang et al.	60.66	90.87	72.76	-	-	-	-
Liu et al.	66.83	75.87	71.07	-	-	-	-
Zhou et al.	72.54	56.44	63.48	-	-	-	-
Liu et al.	7.82	8.14	7.98	-	-	-	-

iterations, and we then use the original training data of each dataset for finetuning. In the pre-processing part, apart from the baseline setting, we also apply color jittering, random cropping, and random rotation with their best ratios as evaluated on the validation dataset for data augmentation. Additionally, the images are trained with a medium setting of the random scale training for maximizing the utilization of the video memory. For the backbone setting, we integrate the ResNext-152-32x8d-FPN-IN5k model, deformable convolution (C4-2), PAN, and MSN modules together to construct a powerful feature extractor. During the proposal generation stage, we adopt deformable PSROI pooling for feature alignment, whereas in the prediction head, we only add an auxiliary character head for mutual promotion using only the ReCTS dataset. Finally, in the post-processing stage, we utilize all effective settings, including polygonal non-maximum suppression, key edge RPP, intermediate model ensemble, and multi-scale ensemble.

The ICDAR 2015 Incidental Scene Text [26] is one of the most popular benchmarks for oriented scene text detection. The images are incidentally captured mostly from streets and shopping malls. Thus, the challenges of this dataset rely on oriented, small, and low-resolution text. This dataset contains 1,000 training samples and 500 testing samples with approximately 2,000 content-recognizable quadri-

lateral word-level bounding boxes. The results of ICDAR 2015 are given in Table 10. From this table, it is clear that our method outperformed all previous methods.

The ICDAR 2017 MLT [27] is the largest multi-lingual (nine languages) oriented scene text dataset, including 7,200 training samples, 1,800 validation samples, and 9,000 testing samples. The challenges associated with this dataset are manifold. Different languages have different annotating styles. For example, most Chinese annotations are long, and there is no specific word interval for sentences. However, most English annotations are short. The annotations of Bangla or Arabic may be frequently entwined with each other, and there is more multi-orientation, perspective distorted text on various complex backgrounds. Furthermore, many images have more than 50 text instances. All instances are well annotated with compact quadrangles. As shown in Table 11, the proposed approach achieved the best performance on the MLT dataset.

ReCTS is the recent ICDAR 2019 Robust Reading Challenge[1] described in Section 4.3.1. Competitors were restricted to submitting at most five results, and all results were evaluated after the deadline. The competition attracted numerous competitors from well-known universities and high-tech companies. The results of the ReCTS are shown in Ta-

---

[1] https://rrc.cvc.uab.es/?ch=12&com=introduction

ble [12]. Our method won first place in the ReCTS detection competition. To clearly evaluate the performance of the final model, we also provide the results of our method on the ReCTS validation set without using a model ensemble. As shown in Table 5, the final model significantly outperformed the baseline by 7.1% in terms of Hmean.

ReCTS End-to-End. One of the main goals of scene text detection is to recognize a text instance [70] that is highly related to the performance of the detection system. To validate the effectiveness and robustness of our detection method, we build a recognition system that incorporate several state-of-the-art methods. Typically, the recognition performance is highly relevant to the quality of the detected boxes. To reveal the precision of our detection, we construct an end-to-end recognition system to demonstrate how our method benefits recognition models. We first crop the images using detected boxes and fed them into four popular recognition models, including decouple attention network [78], convolutional recurrent neural network [79], network of show, attend, read [80], and transformer-based networks [81]. The four models are trained on real samples and 600,000 extra synthetic samples following their default settings for training. The real samples are provided by the official training set, whereas the synthetic samples are synthesized using a render engine [1] and the corpus of the official training set. All images are resized to a specific required height for each recognition model while maintaining the aspect ratio of the original image. In a data batch, all images are padded with white to the maximum width of the images. During the inference stage, we choose the prediction having the highest confidence as the final ensemble result. Both quantitative and qualitative results are presented in Table 12 and Figure 14(b), respectively.

HRSC2016. To demonstrate the generalization ability of our method, we further evaluate its performance on the Level-1 task of the HRSC2016 dataset [77] to demonstrate multi-directional object detection. The ship instances in this dataset are presented in various orientations, and the annotating bounding boxes are based on rotated rectangles. There were 436, 181, and 444 images for training, validating, and testing, respectively. Only the training and validation sets are used for training. The evaluation metric is the same as in [12, 26]. The result is shown in Table 13, showing a significant improvement over the TIoU-Hmean [82]. It also demonstrates the robustness of our method. Qualitative examples of the detection results are shown in Figure 13.

6 Conclusion

In this paper, we have addressed multi-orientation scene text detection using an effective OBD method. Using discretization methodology, OBD, can solve the inconsistent labeling issue by discretizing the point-wise prediction into orderless

Algorithms	R (%)	P (%)	H (%)	TIoU-H (%)	mAP
[15, 83]	-	-	-	-	55.7
[15, 83]	-	-	-	-	69.6
[15, 83]	-	-	-	-	75.7
[15]	-	-	-	-	84.3
Liu et al. [12]	94.8	46.0	61.96	51.1	93.7
Ours	94.1	83.8	**88.65**	**73.3**	89.22
Ours (low cf)	95.7	54.2	69.2	57.5	**94.8**

Table 13: Experimental results for HRSC_2016 dataset. cf: confidence threshold, which is set to 0.01 in the last line.

key edges. To decode accurate vertex positions, we have proposed a simple but effective MTL method to reconstruct the quadrilateral bounding box. Benefiting from OBD, we improve the reliability of the confidence of the bounding box and adopted more effective post-processing methods to improve performance.

Additionally, based on our method, we have conducted thorough ablation studies on six training components, including data arrangement, pre-processing, backbone, proposal generation, prediction head, and post-processing, to explore the potential upper limit of our method. By combining effective modules, we have achieved state-of-the-art results on various benchmarks and won the first place in the recent ICDAR 2019 Robust Reading Challenge on Reading Chinese Text on Signboards. Moreover, using a recognition model, we perform the best in the end-to-end detection and recognition task, verifying that our method is conducive to current recognition methods. To test the generalization ability, we have conducted an experiment on an oriented general object dataset HRSC2016; the results verify that our method can significantly outperform recent state-of-the-art methods.

References

1. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *Int. J. Comput. Vis.*, vol. 116, no. 1, pp. 1–20, 2016. 1, 16

2. L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 3538–3545, IEEE, 2012. 1

3. L. Neumann and J. Matas, "Real-time lexicon-free scene text localization and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1872–1885, 2015. 1

4. L. Neumann and J. Matas, "Efficient scene text localization and recognition with local character refinement," in *Proc. Int. Conf. Doc. Anal. and Recognit.*, pp. 746–750, IEEE, 2015. 1

5. S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. Lim Tan, "Text flow: A unified text detection system in natural scene images," in *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 4651–4659, 2015. 1

6. Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proc. Eur. Conf. Comp. Vis.*, pp. 56–72, Springer, 2016. 1, 14

7. Z. Zhong, L. Jin, S. Zhang, and Z. Feng, "Deeptext: A unified framework for text proposal generation and text detection in natural images," *arXiv preprint arXiv:1605.07314*, 2016. 1

Fig. 13: Qualitative detection results on the HRSC2016 dataset.

8. Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multi-oriented text detection," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 1, 2, 3, 11, 14

9. B. Shi, X. Bai, and S. Belongie, "Detecting oriented text in natural images by linking segments," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 1, 14

10. C. Xue, S. Lu, and F. Zhan, "Accurate scene text detection through border semantics awareness and bootstrapping," in *Proc. Eur. Conf. Comp. Vis.*, pp. 355–372, 2018. 1

11. E. Xie, Y. Zang, S. Shao, G. Yu, C. Yao, and G. Li, "Scene text detection with supervised pyramid context network," *Proc. AAAI Conf. Artificial Intell.*, 2019. 1, 14

12. Y. Liu, S. Zhang, L. Jin, L. Xie, Y. Wu, and Z. Wang, "Omnidirectional scene text detection with sequential-free box discretization," *Proc. Int. Joint Conf. Artificial Intell.*, 2019. 1, 9, 14, 16

13. M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network.," in *Proc. AAAI Conf. Artificial Intell.*, pp. 4161–4167, 2017. 1

14. M. Liao, B. Shi, and X. Bai, "Textboxes++: A single-shot oriented scene text detector," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3676–3690, 2018. 1, 2, 3, 7, 14

15. M. Liao, Z. Zhu, B. Shi, G.-s. Xia, and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 5909–5918, 2018. 1, 2, 14, 16

16. X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan, "Fots: Fast oriented text spotting with a unified network," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 5676–5685, 2018. 1, 14

17. P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li, "Single shot text detector with regional attention," in *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 3047–3055, 2017. 1, 14

18. W. He, X.-Y. Zhang, F. Yin, and C.-L. Liu, "Deep direct regression for multi-oriented scene text detection," *Proc. IEEE Int. Conf. Comp. Vis.*, 2017. 1, 2, 14

19. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 3431–3440, 2015. 2

20. K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 2980–2988, 2017. 2, 4, 6, 12

21. Y. Zhu and J. Du, "Sliding line point regression for shape robust scene text detection," *Proc. Int. Conf. Patt. Recogn.*, 2018. 2

22. C. Xue, S. Lu, and W. Zhang, "Msr: Multi-scale shape regression for scene text detection," *Proc. Int. Joint Conf. Artificial Intell.*, 2019. 2, 11

23. J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, 2018. 2, 14

24. Z. He, Y. Zhou, Y. Wang, S. Wang, X. Lu, Z. Tang, and L. Cai, "An end-to-end quadrilateral regression network for comic panel extraction," in *Proc. ACM Int. Conf. Multimedia*, pp. 887–895, ACM, 2018. 2, 3

25. X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: An efficient and accurate scene text detector," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 2, 7, 14

26. D. Karatzas, L. Gomez-Bigorda, *et al.*, "ICDAR 2015 competition on robust reading," in *Proc. Int. Conf. Doc. Anal. and Recognit.*, pp. 1156–1160, 2015. 2, 7, 8, 12, 14, 15, 16

27. N. Nayef, F. Yin, I. Bizid, H. Choi, Y. Feng, D. Karatzas, Z. Luo, U. Pal, C. Rigaud, J. Chazalon, *et al.*, "Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification-rrc-mlt," in *Proc. Int. Conf. Doc. Anal. and Recognit.*, vol. 1, pp. 1454–1459, IEEE, 2017. 2, 7, 12, 14, 15

28. C.-K. Ch'ng, C. S. Chan, and C.-L. Liu, "Total-text: toward orientation robustness in scene text detection," *Int. J. Doc. Anal. Recognit.*, pp. 1–22, 2019. 2, 8

29. Y. Liu, L. Jin, S. Zhang, C. Luo, and S. Zhang, "Curved scene text detection via transverse and longitudinal sequence connection," *Pattern Recogn.*, vol. 90, pp. 337–345, 2019. 2, 7, 8

30. Y. Sun, Z. Ni, C.-K. Chng, Y. Liu, C. Luo, C. C. Ng, J. Han, E. Ding, J. Liu, D. Karatzas, *et al.*, "ICDAR 2019 Competition on Large-scale Street View Text with Partial Labeling–RRC-LSVT," *Proc. Int. Conf. Doc. Anal. and Recognit.*, 2019. 2, 8

31. C.-K. Chng, Y. Liu, Y. Sun, C. C. Ng, C. Luo, Z. Ni, C. Fang, S. Zhang, J. Han, E. Ding, *et al.*, "ICDAR2019 Robust Reading Challenge on Arbitrary-Shaped Text (RRC-ArT)," *Proc. Int. Conf. Doc. Anal. and Recognit.*, 2019. 2, 8

32. N. Nayef, Y. Patel, M. Busta, P. N. Chowdhury, D. Karatzas, W. Khlif, J. Matas, U. Pal, J.-C. Burie, C.-l. Liu, *et al.*, "ICDAR2019 Robust Reading Challenge on Multi-lingual Scene Text Detection and Recognition–RRC-MLT-2019," *Proc. Int. Conf. Doc. Anal. and Recognit.*, 2019. 2, 8

33. Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 4159–4167, 2016. 2

34. D. Deng, H. Liu, X. Li, and D. Cai, "Pixellink: Detecting scene text via instance segmentation," in *Proc. AAAI Conf. Artificial Intell.*, 2018. 2, 14

35. P. Lyu, C. Yao, W. Wu, S. Yan, and X. Bai, "Multi-oriented scene text detection via corner localization and region segmentation," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 7553–7563, 2018. 2, 14

36. Y. Wu and P. Natarajan, "Self-organized text detection with minimal post-processing via border learning," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 5000–5009, 2017. 2

37. W. Wang, E. Xie, X. Li, W. Hou, T. Lu, G. Yu, and S. Shao, "Shape Robust Text Detection with Progressive Scale Expansion Network," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2019. 2, 14

38. T. He, W. Huang, Y. Qiao, and J. Yao, "Text-attentional convolutional neural network for scene text detection," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2529–2541, 2016. 2

39. Y. Zhu, J. Du, and X. Wu, "Adaptive period embedding for representing oriented objects in aerial images," *IEEE Trans. Geoscience & Remote Sensing*, 2020. 7

40. C.-K. Chng and C.-S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *Proc. Int. Conf. Doc. Anal. and Recognit.*, pp. 935–942, 2017. 8

41. B. Shi, C. Yao, M. Liao, M. Yang, P. Xu, L. Cui, S. Belongie, S. Lu, and X. Bai, "ICDAR2017 competition on reading chinese text in the wild (RCTW-17)," in *Proc. Int. Conf. Doc. Anal. and Recognit.*, vol. 1, pp. 1429–1434, 2017. 8

42. D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. I. Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazàn, and L. P. D. L. Heras, "Icdar 2013 robust reading competition," in *Proc. Int. Conf. Doc. Anal. and Recognit.*, pp. 1484–1493, 2013. 8

43. C. Yao, X. Bai, W. Liu, and Y. Ma, "Detecting texts of arbitrary orientations in natural images," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 1083–1090, 2012. 8

44. A. Veit, T. Matera, *et al.*, "Coco-text: Dataset and benchmark for text detection and recognition in natural images," in *arXiv preprint arXiv:1601.07140*, 2016. 8

45. X.-C. Yin, W. Y. Pei, J. Zhang, and H. W. Hao, "Multi-orientation scene text detection with adaptive clustering," in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, p. 1930, 2015. 8

46. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2017. 11

47. J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 764–773, 2017. 11, 12

48. A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 6399–6408, 2019. 11

49. Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring r-cnn," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 6409–6418, 2019. 11, 12

50. S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proc. Advances in Neural Inf. Process. Syst.*, pp. 91–99, 2015. 11, 12

51. L. Xie, Y. Liu, L. Jin, and Z. Xie, "DeRPN: Taking a further step toward more general object detection," in *Proc. AAAI Conf. Artificial Intell.*, vol. 33, pp. 9046–9053, 2019. 11

52. J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 7263–7271, 2017. 11

53. Y. Li, Y. Chen, N. Wang, and Z. Zhang, "Scale-aware trident networks for object detection," *Proc. IEEE Int. Conf. Comp. Vis.*, 2019. 11

54. J. Wang, K. Chen, S. Yang, C. C. Loy, and D. Lin, "Region proposal by guided anchoring," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 2965–2974, 2019. 11

55. J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Proc. Advances in Neural Inf. Process. Syst.*, pp. 379–387, 2016. 12

56. Q. Yang, M. Cheng, W. Zhou, Y. Chen, M. Qiu, W. Lin, and W. Chu, "Inceptext: A new inception-text module with deformable psroi pooling for multi-oriented scene text detection," *Proc. Int. Joint Conf. Artificial Intell.*, 2018. 12

57. A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 761–769, 2016. 12

58. H. Hu, C. Zhang, Y. Luo, Y. Wang, J. Han, and E. Ding, "Wordsup: Exploiting word annotations for character based text detection," *Proc. IEEE Int. Conf. Comp. Vis.*, 2017. 14

59. Y. Xu, Y. Wang, W. Zhou, Y. Wang, Z. Yang, and X. Bai, "Textfield: Learning a deep direction field for irregular scene text detection," *IEEE Trans. Image Process.*, 2019. 14

60. J. Tang, Z. Yang, Y. Wang, Q. Zheng, Y. Xu, and X. Bai, "Detecting dense and arbitrary-shaped scene text by instance-aware component grouping," *Pattern Recogn.*, 2019. 14

61. C. Zhang, B. Liang, Z. Huang, M. En, J. Han, E. Ding, and X. Ding, "Look More Than Once: An Accurate Detector for Text of Arbitrary Shapes," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2019. 14

62. Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character Region Awareness for Text Detection," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 9365–9374, 2019. 14

63. Z. Huang, Z. Zhong, L. Sun, and Q. Huo, "Mask r-cnn with pyramid attention network for scene text detection," in *Proc. Winter Conf. Appl. of Comp. Vis.*, pp. 764–772, IEEE, 2019. 14

64. Z. Zhong, L. Sun, and Q. Huo, "Improved localization accuracy by locnet for faster r-cnn based text detection in natural scene images," *Pattern Recogn.*, vol. 96, p. 106986, 2019. 14

65. T. He, Z. Tian, W. Huang, C. Shen, Y. Qiao, and C. Sun, "An end-to-end textspotter with explicit alignment and attention," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 5020–5029, 2018. 14

66. Y. Liu, L. Jin, and C. Fang, "Arbitrarily shaped scene text detection with a mask tightness text detector," *IEEE Trans. Image Process.*, 2019. 14

67. S. Long, J. Ruan, W. Zhang, X. He, W. Wu, and C. Yao, "Textsnake: A flexible representation for detecting text of arbitrary shapes," in *Proc. Eur. Conf. Comp. Vis.*, pp. 20–36, 2018. 14

68. W. He, X.-Y. Zhang, F. Yin, Z. Luo, J.-M. Ogier, and C.-L. Liu, "Realtime multi-scale scene text detection with scale-based region proposal network," *Pattern Recogn.*, vol. 98, p. 107026, 2020. 14

69. P. Lyu, M. Liao, C. Yao, W. Wu, and X. Bai, "Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes," in *Proc. Eur. Conf. Comp. Vis.*, pp. 67–83, 2018. 14

70. H. Xie, S. Fang, Z.-J. Zha, Y. Yang, Y. Li, and Y. Zhang, "Convolutional attention networks for scene text recognition," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 15, no. 1s, pp. 1–17, 2019. 14, 16

71. M. Liao, P. Lyu, M. He, C. Yao, W. Wu, and X. Bai, "Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019. 14

72. W. Wang, E. Xie, X. Song, Y. Zang, W. Wang, T. Lu, G. Yu, and C. Shen, "Efficient and Accurate Arbitrary-Shaped Text Detection with Pixel Aggregation Network," *Proc. IEEE Int. Conf. Comp. Vis.*, 2019. 14

73. X. Wang, Y. Jiang, Z. Luo, C.-L. Liu, H. Choi, and S. Kim, "Arbitrary shape scene text detection with adaptive text region representation," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2019. 14

74. S. Qin, A. Bissacco, M. Raptis, Y. Fujii, and Y. Xiao, "Towards Unconstrained End-to-End Text Spotting," *Proc. IEEE Int. Conf. Comp. Vis.*, 2019. 14

75. F. Wei, H. Wenhao, Y. Fei, Z. Xu-Yao, and C.-L. Liu, "TextDragon: An End-to-End Framework for Arbitrary Shaped Text Spotting," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2019. 14

76. Z. Zhong, L. Sun, and Q. Huo, "An anchor-free region proposal network for faster r-cnn-based text detection approaches," *Int. J. Document Analysis & Recogn.*, vol. 22, no. 3, pp. 315–327, 2019.

14

77. Z. Liu, J. Hu, L. Weng, and Y. Yang, "Rotated region based cnn for ship detection," in *Proc. IEEE Int. Conf. Image Process.*, pp. 900–904, IEEE, 2017. 14, 16

78. W. Tianwei, Z. Yuanzhi, J. Lianwen, C. Luo, X. Chen, Y. Wu, Q. Wang, and M. Cai, "Decoupled attention network for text recognition," in *Proc. AAAI Conf. Artificial Intell.*, 2020. 16

79. B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, 2017. 16

80. H. Li, P. Wang, C. Shen, and G. Zhang, "Show, attend and read: A simple and strong baseline for irregular text recognition," in *Proc. AAAI Conf. Artificial Intell.*, 2019. 16

81. P. Wang, L. Yang, H. Li, Y. Deng, C. Shen, and Y. Zhang, "A Simple and Robust Convolutional-Attention Network for Irregular Text Recognition," *arXiv:1904.01375*, 2019. 16

82. Y. Liu, L. Jin, Z. Xie, C. Luo, S. Zhang, and L. Xie, "Tightness-aware evaluation protocol for scene text detection," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 9612–9620, 2019. 16

83. R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 1440–1448, 2015. 16

(a) Detection only results on MLT dataset.

(b) End-to-end results on ReCTS.

Fig. 14: Visualization of the qualitative results outputted by the proposed approach.