Rotation-Sensitive Regression for Oriented Scene Text Detection

Minghui Liao[1], Zhen Zhu[1], Baoguang Shi[1], Gui-song Xia[2], Xiang Bai[1*]

[1]Huazhong University of Science and Technology

[2]Wuhan University

{mhliao, zzhu}@hust.edu.cn, shibaoguang@gmail.com, guisong.xia@whu.edu.cn, xbai@hust.edu.cn

Abstract

Text in natural images is of arbitrary orientations, requiring detection in terms of oriented bounding boxes. Normally, a multi-oriented text detector often involves two key tasks: 1) text presence detection, which is a classification problem disregarding text orientation; 2) oriented bounding box regression, which concerns about text orientation. Previous methods rely on shared features for both tasks, resulting in degraded performance due to the incompatibility of the two tasks. To address this issue, we propose to perform classification and regression on features of different characteristics, extracted by two network branches of different designs. Concretely, the regression branch extracts rotation-sensitive features by actively rotating the convolutional filters, while the classification branch extracts rotation-invariant features by pooling the rotation-sensitive features. The proposed method named Rotation-sensitive Regression Detector (RRD) achieves state-of-the-art performance on three oriented scene text benchmark datasets, including ICDAR 2015, MSRA-TD500, RCTW-17 and COCO-Text. Furthermore, RRD achieves a significant improvement on a ship collection dataset, demonstrating its generality on oriented object detection.

1. Introduction

Reading text in the wild is an active research field in computer vision, driven by many real-world applications such as license plate recognition [33], guide board recognition [37], and photo OCR [3]. A scene text reading system generally begins with localizing text regions on which the recognition is then performed. Consequently, one of the main bottleneck of such a system lies in the quality of text detection.

Despite the great success of recent general object detection algorithms [9, 8, 36, 34, 28, 35], scene text detection remains challenging mainly due to arbitrary orientations,

(a) Input image   (b) Shared feature   (c) Result

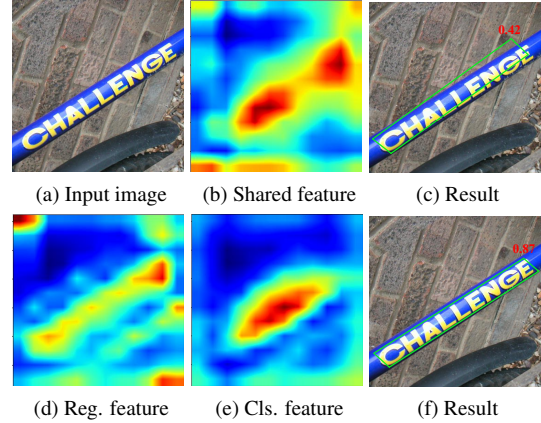(d) Reg. feature   (e) Cls. feature   (f) Result

Figure 1: Visualization of feature maps and results of baseline and RRD. Red numbers are the classification scores. (b): the shared feature map for both regression and classification; (c): the result of shared feature; (d) and (e): the regression feature map and classification feature map of RRD; (f): the result of RRD.

small sizes, and significantly varied aspect ratios of text in natural images. In fact, general object detection methods usually focus on detecting objects in terms of horizontal bounding boxes, which are accurate enough for most objects such as person, vehicles, etc. Yet, horizontal bounding box is not appropriate for representing long and thin objects in arbitrary orientations (see Fig. 1a). Text in natural images is a typical example of multi-oriented long and thin object, which is better covered by oriented bounding boxes. Directly applying general object detection methods to scene text detection would generally lead to poor performance.

Some recent scene text detectors [26, 29] successfully adopt general object detection methods to scene text detection with some dedicated designs, yielding a great improvement in scene text detection. Generally, recent object detection consists of predicting object category (*i.e.,* classification) and regressing bounding box for accurate localization. Both tasks rely on shared features which are rotation-invariant attributing to the use of pooling layers

---

*Corresponding author.

in general convolutional neural network (CNN) architecture [4, 23, 18]. This pipeline is also adopted in recent scene text detectors inspired by general object detection methods. Although it is well known that rotation-invariant features can boost the performance of classification, the rotation invariance is not beneficial for regressing arbitrary oriented bounding boxes. This conflicting issue between classification and regression may not be very important on oriented objects with limited aspect ratio. However, unlike Latin text, there is not a "blank" between neighbor words in non-Latin text such as Chinese, Arabic, Japanese, etc, which possess long text lines frequently and are often detected at the line level instead of word spotting. In this sense, detecting long oriented text lines is obviously a non-trivial task that satisfies with the practical requirements of a more general text reading system for multi-lingual text. Thus, for scene text, especially non-Latin text lines which are usually long and thin, and of arbitrary orientations, using rotation-invariant features would hinder the regression of such oriented bounding boxes. Another important issue for detecting arbitrary oriented long text having extreme aspect ratios is the requirement of more flexible receptive field.

To alleviate the above-mentioned issues of arbitrary oriented scene text detection, we propose to separate the regression task from the classification task. More specifically, as depicted in Fig. 2, the proposed method named Rotation-sensitive Regression Detector (RRD) performs classification with rotation-invariant features, and oriented bounding box regression with rotation-sensitive features. For that, we adopt oriented response convolution [52] instead of normal convolution in the network architecture. The rotation-invariant features for classification are then obtained by an oriented response pooling layer [52]. To the best of our knowledge, this is the first time to apply oriented response network to object detection task. In addition, we also propose an inception block of three-scale convolutional kernels to give flexible receptive field better covering long text. As shown in Fig. 1, the regression feature map (Fig. 1d) of RRD contains richer and more precise orientation information and its classification feature map (Fig. 1e) is more intensive, compared with the conventional shared feature map (Fig. 1b) for both classification and regression. The final results (Fig. 1c and Fig. 1f) also demonstrate the quality of the feature maps.

The main contributions of this paper are three folds: 1) We propose a novel idea of using rotation-sensitive features for regressing oriented bounding boxes while using rotation-invariant features for classification. This separation gives rise to a more accurate regression in detecting arbitrary oriented long and thin objects; 2) A general framework for arbitrary oriented object (*e.g.*, scene text) detection is proposed. It can be easily embedded into any existing detection architectures, improving the performance without

obvious loss of speed. 3) The proposed RRD is also an effective and efficient oriented scene text detector, with the generality of detecting both Latin and non-Latin text.

2. Related Work

2.1. Object Detection

Recent object detectors [9, 8, 36, 34, 28, 6, 35, 27] leverage the powerful learning ability of CNN to detect objects, and achieve impressive performances. Generally, most CNN-based object detectors share a common pipeline which consists of object classification predicting the object category and bounding box regression for accurate localization. Both classification and regression rely on shared translation and rotation-invariant features attributing to pooling layers involved in classical CNN architecture. Although these invariances are beneficial and important for classification task, accurate regression requires translation and rotation-sensitive features. In this sense, classification and regression are somewhat incompatible to each other about the demands of translation invariance [6] and rotation invariance. R-FCN [6] introduces sensitive ROI-pooling to alleviate the translation invariance problem. In [35, 27], the authors propose to fuse the high-resolution and low-resolution feature maps to balance the conflict. To the best of our knowledge, the rotation invariance incompatibility has never been explicitly considered for object detection. This is particularly important for regressing oriented bounding boxes, which are more appropriate for detecting arbitrary oriented long and thin objects. This paper focuses on a typical example of such object detection, i.e., scene text detection, by explicitly introducing rotation-sensitive features to the CNN pipeline.

2.2. Scene Text Detection

Different from scene text proposal methods [17, 2] which mainly concern the recall, scene text detection methods output much less output bounding boxes, considering the trade-off between recall and precision. Recently, numerous inspiring ideas and promising methods [48, 49, 43, 19, 50, 11, 44, 26, 51, 38, 29, 13, 12, 14, 46, 5, 24, 42] have been proposed. A great improvement has been achieved compared to traditional methods [15, 32, 16]. Based on the representation of detection output, scene text detectors could be roughly divided into two categories: 1) Horizontal text detectors [49, 19, 11, 11, 44, 26] which detect words or text lines in terms of horizontal bounding boxes in the same way as general object detectors. This implies that horizontal scene text detection may benefit from the developments of general object detection such as [34, 28]. The works in [11, 26] are such examples; 2) Multi-oriented text detectors [48, 50, 51, 38, 29, 13, 12, 14, 46] which focus on detecting text of arbitrary orientations. The detection out-

(a) Backbone

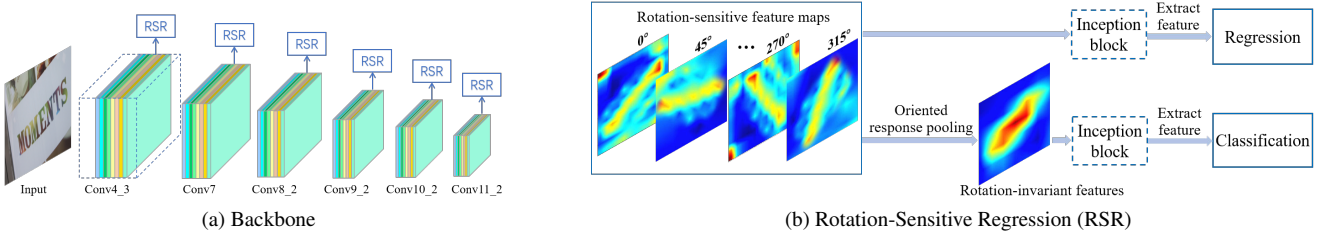(b) Rotation-Sensitive Regression (RSR)

Figure 2: Architecture of RRD. (a) The rotation-sensitive backbone follows the main architecture of SSD while changing its convolution into oriented response convolution. (b) The outputs of rotation-sensitive backbone are rotation-sensitive feature maps, followed by two branches: one for regression and another for classification based on oriented response pooling. Note that the inception block is optional.

puts are usually represented by either oriented rectangles or more generally quadrilaterals enclosing arbitrary oriented words or text lines. Compared to horizontal bounding box representation, additional variables such as the angle or vertex coordinates are required for representing multi-oriented text bounding boxes.

All the modern scene text detectors inspired by recent CNN-based general object detectors use shared features for both classification and regression. Compared to these modern multi-oriented scene text detectors, this paper proposes to explicitly use rotation-sensitive CNN features for oriented bounding box regression while adopting rotation-invariant features for classification. This results in a more accurate oriented bounding box regression for arbitrary oriented text, especially for the long words or text lines.

2.3. Rotation-sensitive CNN Features

Rotation-invariant features are important for a robust classification. Modern CNN architectures usually involve pooling layers achieving rotation invariance to a certain extent. Some recent works [10, 7, 52] focus on enhancing the rotation invariance of the CNN features to further improve the classification performance. For example, ORN [52] proposes to actively rotate during convolution, producing rotation-sensitive feature maps. This is followed by an oriented pooling operation, giving rise to enhanced rotation invariance, and thus resulting in better classification performance. The proposed RRD is inspired by ORN [52]. The enhanced rotation-invariant features are used for text presence prediction. We propose to adopt rotation-sensitive features to regress oriented bounding boxes, yielding accurate detection of arbitrary oriented long objects (*e.g.*, scene text). To the best of our knowledge, this is the first time that explicit rotation-sensitive features are used for arbitrary oriented object detection.

3. Rotation-Sensitive Regression Detector

3.1. Overview

RRD is an end-to-end trainable, fully convolutional neural network whose architecture is inspired by SSD [28]. Its

architecture is illustrated in Fig. 2, which uses VGG16 [40] as its backbone network, with extra layers added in the same manner as SSD. As shown in Fig. 2, six layers of the backbone network are taken for dense prediction.

The dense prediction is similar to that of SSD [28]. For every default box [28], RRD classifies its label (text or non-text) and regresses relative offsets. After that, RRD applies the offsets to the default boxes classified as positive, producing a number of quadrilaterals, each with a score. The quadrilaterals are filtered by non-maximum suppression, which outputs final detections.

The key novelty of our method is the dense prediction part, where RRD extracts two types of feature maps of different characteristics for classification and regression respectively. The feature map for classification is insensitive to text orientation, while the feature map for regression is sensitive. As mentioned before, these characteristics well fit the nature of the two tasks.

3.2. Rotation-Sensitive Regression

Text coordinates are sensitive to text orientation. Therefore, the regression of coordinate offsets should be performed on rotation-sensitive features. Oriented response convolution encodes the rotation information by actively rotating its convolutional filters, producing rotation-sensitive features for regression.

Different from standard CNN features, RRD extracts rotation-sensitive features with active rotating filters (ARF) [52]. An ARF convolves a feature map with a *canonical filter* and its rotated clones. The filters are rotated following the method in [52]. Denote the canonical filter of ARF as $F_0 \in \Re^{k \times k \times N}$, where k is the kernel size, N is the number of rotations. ARF makes $N-1$ clones of the canonical filter by rotating it to different angles, respectively $F_j, j = 1 : N$. Let $M_i(j)$ and $M_o(j)$ denote the input feature map and the output feature map of j-th orientation respectively. ARF convolves a feature map by computing:

$$M_o(j) = \sum_{n=0}^{N-1} F_j(n) * M_i(n), j = 0, ..., N-1, \quad (1)$$

3

where $F_j(n)$ indicates the n-th orientation channel of F_j. After convolution, it produces a response map of N channels, each corresponding to the response of the canonical filter or its rotated clone. N is set to 8 in practice.

ARF produces extra channels to incorporate richer rotation information. With the help of ARF, ORN produces feature maps with orientation channels, capturing rotation-sensitive features and improving its generality for rotated samples which has never seen before. Besides, since the parameters between the N filters are shared, learning ARF requires much less training examples.

In addition, in order to make the receptive field suitable for long text lines, we adopt inception blocks for both branches. The inception block concatenates the output feature maps produced by three filters of different sizes. The filter sizes are respectively $m \times m$, $m \times n$ and $n \times m$, where m is set to 3 and n is set to $(9, 7, 5)$ in the first, the second, and the last stages respectively. Inception blocks result in receptive fields of different aspect ratios. They are particularly helpful for detecting long text. Therefore, they are used in line-based text detection, but discarded in word-based text detection, where long text is rare.

3.3. Rotation-Invariant Classification

In contrast to regression, the classification of text presence should be rotation-invariant, *i.e.*, text regions of arbitrary orientations should be classified as positive. Therefore, a rotation-invariant feature map should be extracted for this task.

ORN achieves rotation invariance by pooling responses of all N response maps. As shown in Fig. 2b, the rotation-sensitive feature maps are pooled along their depth axis. Assuming that M_{or} is a rotation-sensitive input feature map of N orientation channels, the rotation-invariant feature map $M_{pooling}$ is an element-wise max of M_{or} with the index of orientations, which can be calculated as follows:

$$M_{pooling} = \max_{k=0}^{N-1} M_{or}(k), \qquad (2)$$

Since the pooling operation is orderless and applied to all N response maps, the resulting feature map is locally invariant to object rotation. Therefore, we use this feature map for classification. Besides, the setting of inception block is the same as the regression branch.

3.4. Default Boxes and Prediction

The default boxes are horizontal rectangles with different sizes and aspect ratios. Let $\mathbf{B}_0 = (x_0, y_0, w_0, h_0)$ denote a horizontal default box, which can also be represented by its four vertexes $\mathbf{Q}_0 = (v_1^0, v_2^0, v_3^0, v_4^0)$, where $v_i^0 = (x_i^0, y_i^0), i \in \{1, 2, 3, 4\}$.

The regression branch predicts offsets from a default box to a quadrilateral. A quadrilateral is described as $\mathbf{Q} =$

(v_1, v_2, v_3, v_4), where $v_i = (x_i, y_i), i \in \{1, 2, 3, 4\}$ are four vertexes of a quadrilateral. For each default box, the prediction layer outputs the classification scores and offsets $(\Delta x_1, \Delta y_1, \Delta x_2, \Delta y_2, \Delta x_3, \Delta y_3, \Delta x_4, c)$ between the default box $\mathbf{Q}_0$ and the bounding box result $\mathbf{Q}$. The final output quadrilateral is encoded with the corresponding default box:

$$
\begin{aligned}
x_i &= x_i^0 + w_0 \Delta x_i, i = 1, 2, 3, 4, \\
y_i &= y_i^0 + h_0 \Delta y_i, i = 1, 2, 3, 4,
\end{aligned} \qquad (3)
$$

where w_0 and h_0 denote the width and height of the default box respectively. In addition, a non-maximum suppression with quadrilaterals is applied in the prediction period.

3.5. Training

Ground Truth. The ground truth of an oriented text region can be described as a quadrilateral $\mathbf{G}_q = (v_1, v_2, v_3, v_4)$, where $v_i = (x_i, y_i), i \in \{1, 2, 3, 4\}$ are the vertices of the quadrilateral. We argue that careful selection of the first point in quadrilateral is helpful for regression. Thus, we follow a scheme in [25], which determine the first point based on the distances from its corresponding maximum horizontal bounding box.

Loss Function. In the training phase, default boxes are matched to the ground-truth boxes according to box overlap following the match scheme in [28]. For efficiency, the minimum horizontal rectangle enclosing the quadrilateral is used in the matching period.

We adopt a similar loss function to the one used in [28]. More specifically, let x be the match indication matrix. For the i-th default box and the j-th ground truth, $x_{ij} = 1$ means a match following the box overlap between them, otherwise $x_{ij} = 0$. Let c be the confidence, l be the predicted location, and g be the ground-truth location. The loss function is defined as:

$$L(x, c, l, g) = \frac{1}{N}(L_{\text{cls}}(x, c) + \alpha L_{\text{reg}}(x, l, g)), \qquad (4)$$

where N indicates the number of default boxes that match ground-truth boxes, and α is set to 0.2 for quick convergence. We adopt a smooth L1 loss [8] for L_{reg} and a 2-class softmax loss for L_{cls}. We follow the same online hard negative mining strategy as [28].

4. Experiments

4.1. Datasets

We conduct extensive experiments to verify the effectiveness of RRD in multiple aspects. Altogether, seven datasets are used in the experiments. We first evaluate RRD on RCTW-17 [39] and MSRA-TD500 [48] to show its effectiveness at detecting long and oriented text. Both

4

datasets contain many instances of such, and they are annotated in text lines. To further assess the importance of using rotation-sensitive features for regressing long and oriented boxes, we construct a subset of RCTW-17 by picking text instances with extreme aspect ratios. COCO-Text dataset [45] is evaluated to prove the accurate of regression, which provides a evaluation protocol where the IOU threshold is set to 0.75. Then we test RRD on ICDAR 2015 incidental text dataset [20], which contains oriented English words. The ICDAR 2013 focused text dataset [21] is also evaluated, showing its good performance on horizontal text detection. In the end, in order to show the generality of RRC, we evaluate RRD on HRSC2016 [31], a dataset of high resolution ship collection.

Reading Chinese Text in the Wild (RCTW-17) contains 12,000 images taken from streets, screen shots and indoor scenes etc. The sizes of images range from small to extremely large; Since Chinese words are not separated by blank spaces, long text lines are common.

RCTW-Long is a sub-dataset drawn from RCTW-17 featuring long text. The training set of RCTW-Long consists of 1323 images picked from the original training set, and the test set 537 from the original test set. Specifically, a bounding box is defined as a long box if its aspect ratio is greater than t or less than $1/t$, otherwise a short box. t is set to 5 for the training set and 7 for the test set. We only select the images with more long boxes than short boxes for constructing this dataset.

MSRA-TD500 contains 500 natural images taken by pocket cameras from indoor and outdoor scenes. The dataset is divided into 300 training images and 200 test images. This dataset contains both English and Chinese text. Compared to RCTW-17, this dataset contains less text instances per image, but larger variance in text orientations.

ICDAR 2015 Incidental Text (IC15) comes from the Challenge 4 of ICDAR 2015 Robust Reading Competition. Images of this dataset were captured by Google Glasses in streets, shopping malls, etc., in an incidental manner. Consequently, many images of this dataset are of low resolution, and text is in various orientations. IC15 contains 1,000 training images and 500 test images. Annotations are provided in terms of word bounding boxes.

COCO-Text is a large dataset which contains 63686 images, where 43,686 of the images is used for training, 10,000 for validation, and 10,000 for testing. It is one of the challenges of ICDAR 2017 Robust Reading Competition.

ICDAR 2013 Focused Text (IC13) is composed of 229 training images and 233 testing images. This dataset contains only horizontal and focused text. Images in IC13 are in high resolutions.

High Resolution Ship Collection 2016 (HRSC2016) contains 1061 images divided into 436, 181, 444 images for training, validation, test set, respectively. The images are from two scenarios including ships on sea and ships close inshore derived from Google Earth. Ships are abundantly labeled with rotated polygons along with some extra information such as ship types and ship head locations etc.

4.2. Implementation Details

RRD is optimized by the ADAM [22] algorithms on all datasets. For all scene text datasets, RRD is pre-trained on SynthText [11] for 30k iterations and fine-tuned on real data. In the first 5–10k iterations, images are resized to 384×384 after random cropping. The learning rate is fixed to 10^{-4}. Another 5–10k iterations is followed, where images are resized to 768×768 and the learning rate decayed to 10^{-5}. The aspect ratios of default boxes are set to $1, 2, 3, 5, 1/2, 1/3, 1/5$ for word-based dataset (IC15 and IC13), and $1, 2, 3, 5, 7, 9, 15, 1/2, 1/3, 1/5, 1/7, 1/9, 1/15$ for text line-based dataset (RCTW-17, RCTW-Long, and MSRA-TD500).

For HRSC2016 experiments, we scale input images to 384×384 due to lots of thin and long ship instances. We train the model for around 15k iterations on three NVIDIA TITAN Xp GPUs from scratch at the learning rate of 10^{-4} with the batch size set to 32. Then, we continue training for about 1k iterations with learning rate decayed to 10^{-5} while keeping other settings unchanged. The default boxes settings remain the same as the line-based scene text datasets.

4.3. Ablation Study

We apply several variants of RRD to verify the effectiveness of rotation-sensitive regression. The tested variants of RRD are summarized as follows:
Baseline: architecture without inception block, using shared conventional feature maps for both regression and classification; **Baseline+inc:** baseline architecture using inception blocks; **Baseline+inc+rs:** architecture with inception block, using rotation-sensitive features for both regression and classification; **Baseline+inc+rs+rotInvar:** the proposed RRD. Note that for word-based datasets, inception block is not applied and we also name it RRD.

Method	Recall	Precision	F-measure
Baseline	0.6880	0.6925	0.6902
Baseline+inc	0.7102	0.8018	0.7532
Baseline+inc+rs	0.7423	**0.8979**	0.8127
Baseline+inc+rs+rotInvar (RRD)	**0.8146**	0.8535	**0.8336**

Table 1: Evaluation results of several variants of RRD on RCTW-Long dataset.

Oriented long text needs more accurate bounding boxes. Given the examples in Fig. 5 , the short text can be easily covered by a bounding box even though its orientation is not

Figure 3: Some results of RRD on RCTW-17 (first row) and IC15 (second row).

(a) Results of Baseline+inc

(b) Results of RRD

Figure 4: Some results on images in RCTW-Long dataset.

Figure 5: Red boxes: ground truths; Yellow boxes: detections. The corresponding angle is the same while the IOU diverges hugely.

accurate, while the long text is much more sensitive to the orientation. Thus, we conduct ablation study on RCTW-Long dataset, a typical dataset which mainly consists of long and oriented text, to verify the superiority of rotation-sensitive regression. The quantitative comparison of several variants of RRD described above is shown in Tab. 1. Following standard evaluation protocol on RCTW dataset, we traverse the threshold of detection scores from 0.1 to 1 with a step of 0.02 to get the best F-measure for all models. The

detailed comparison is given in the following.

Inception block. Compared with the baseline model, the inception block achieves an improvement of about 6 percents. This implies that the inception block can effectively alleviate the limited receptive field problem for detecting long and thin text.

Rotation-sensitive regression and classification. We also test the architecture using rotation-sensitive features for both regression and classification. Such model "Baseline+inc+rs" further improves the "Baseline+inc" architecture by about 6 percents. Therefore, it is evident that the rotation-sensitive features are useful for long, thin and oriented text detection.

Rotation-sensitive regression and Rotation-invariant classification. The last model in Tab. 1 is the proposed RRD which uses rotation-sensitive and rotation-invariant features for regression and classification respectively. It alleviates the dilemma between regression and classification, outperforming all other models in Tab 1 by a large margin.

Some qualitative comparisons between the proposed RRD and its variant "Baseline+inc" are illustrated in Fig 4. As shown, the proposed RRD yields more convincing classification scores and more accurate bounding boxes. Specifically, the bounding boxes having scores lower than 0.5 are discarded in practice. Consequently, some text bounding boxes generated by "Baseline+inc" are discarded. Whereas, all text bounding boxes given by RRD are reserved. This qualitative comparison also shows the effectiveness of RRD thanks to the specialized features for regression and classification, alleviating the incompatibility of the two tasks.

4.4. Results on Scene Text Benchmarks

RCTW-17. RCTW-17 is a large line-based dataset which mainly consists of Chinese text. Thus, the inception block is important. The quantitative results using the official evaluation scheme is depicted in Tab. 2. RRD outperforms the

RCTW-17					ICDAR2015				
Methods	Recall	Precision	F-measure	FPS	Methods	Recall	Precision	F-measure	FPS
Official baseline [39]	0.404	**0.76**	0.528	8.9	Zhang et al. [50]	0.43	0.71	0.54	-
EAST-ResNet*	0.478	0.597	0.531	7.4	Tian et al. [44]	0.52	0.74	0.61	7.1
Baseline+inc	**0.459**	0.659	0.541	**10.6**	Shi et al. [38]	0.768	0.731	0.750	8.9
RRD	0.453	0.724	**0.557**	10	Liu et al. [29]	0.682	0.732	0.706	-
Baseline+inc+MS	**0.595**	0.744	0.661	-	Zhou et al. [51]	0.735	0.836	0.782	**13.2**
RRD+MS	0.591	**0.775**	**0.670**	-	He et al. [12]	0.73	0.80	0.77	-
MSRA-TD500					Hu et al. [14]	0.77	0.793	0.782	-
Methods	Recall	Precision	F-measure	FPS	**Baseline**	0.762	**0.871**	0.813	8.5
Zhang et al. [50]	0.67	0.83	0.74	0.48	**RRD**	**0.79**	0.856	**0.822**	6.5
He et al. [13]	0.7	0.77	0.74	1.1	Zhou et al. MS [51]	0.783	0.833	0.807	-
Shi et al. [38]	0.7	0.86	0.77	8.9	Hu et al. MS [14]	0.77	0.793	0.782	-
Zhou et al. [51]	0.67	**0.87**	0.76	**13.2**	He et al. MS [13]	0.80	0.82	0.81	-
Baseline+inc	0.69	0.79	0.74	10.6	**Baseline+MS**	0.785	0.878	0.828	-
RRD	**0.73**	**0.87**	**0.79**	10	**RRD+MS**	**0.8**	**0.88**	**0.838**	-

Table 2: Text detection results on multi-oriented scene text benchmarks: MSRA-TD500, RCTW-17, and IC15.

official baseline based on SegLink [38] by about 3.3 percents in terms of F-measure. Even though EAST-ResNet [1] uses a stronger backbone (ResNet), RRD still performs better, achieving 2.6 percents F-measure improvement. Moreover, using multi-scale inputs including $384 \times 384, 384 \times 768, 768 \times 384, 768 \times 768, 1024 \times 1024, 1536 \times 1536$, "RRD+MS" further achieves 11.3 percents improvement than RRD using single-scale input. Some qualitative results of RRD on this dataset are given in Fig. 3.

MSRA-TD500. MSRA-TD500 is also a line-based dataset which contains Chinese and English. Thus, the inception block is applied for this dataset. As shown in Tab. 2, RRD outperforms the state-of-the-art methods in terms of precision, recall, and F-measure. To further verify the effectiveness of rotation-sensitive regression, we also evaluate the "Baseline+inc" architecture, which degrades the performance of RRD by 5 percents in terms of F-measure.

COCO-Text. The experiments on COCO-Text Challenge prove the accurate regression of RRD. As shown in Tab. 3, COCO-Text Challenge provides a more strict evaluation protocol in which the IOU threshold is set to 0.75, where RRD has a huge superiority because the detection results of RRD is much more accurate, benefiting from the rotation-sensitive regression. Specifically, RRD achieves comparable results with the previous state-of-the-art method when the IOU threshold is set to 0.5 while outperforms all the previous methods by at least 6.1 percents when the IOU threshold is set to 0.75.

Methods	IOU=0.5			IOU=0.75		
	R	P	F	R	P	F
UM [1]	**0.66**	0.48	0.55	0.31	0.23	0.26
TDN SJTU v2 [1]	0.54	0.62	0.58	0.28	0.32	0.30
Text Detection DL [1]	0.62	0.60	**0.61**	0.26	0.25	0.25
RRD+MS	0.57	**0.64**	**0.61**	**0.34**	**0.38**	**0.36**

Table 3: Experimental results on COCO-Text Challenge.

IC15. For the comparison of RRD with the state-of-the-art results on IC15 dataset, we use the scale of 1024×1024 for single scale testing in RRD. The multi-scale testing includes the scales of $384 \times 384, 768 \times 768, 1024 \times 1024$, and 1536×1536. The comparison with the state-of-the-art results on IC15 dataset is given in Tab. 2. Even though IC15 is a word-based dataset whose text bounding boxes are not extremely long, RRD still achieves about 1 percent performance gain than the baseline with both single scale and multi-scale settings. Furthermore, RRD outperforms the state-of-the-art results by 4 percents with single scale setting and 2.8 percents in the case of multi-scale setting.

Method	Recall	Precision	F-measure
FCRNall+filts [11]	0.76	0.92	0.83
TextBoxes [26]	0.74	0.88	0.81
TextBoxes+MS [26]	0.83	0.89	0.86
Seglink [38]	0.83	0.88	0.85
Tian et al. [44]	0.83	**0.93**	0.88
Tang et al. [41]	0.87	0.92	**0.90**
He et al. [12]	0.86	0.89	0.88
He et al. [13]	0.81	0.92	0.86
WordSup+MS [14]	**0.88**	0.93	**0.90**
Baseline	0.74	0.88	0.81
Baseline+MS	0.85	0.92	0.88
RRD	0.75	0.88	0.81
RRD+MS	0.86	0.92	0.89

Table 4: Experimental results on IC13. MS stands for multi-scale testing.

IC13. Although the proposed RRD is specifically designed for multi-oriented text detection, we also evaluate RRD on IC13 dataset consisting of horizontal text. The experimental results are depicted in Tab. 4, showing that the proposed scheme has no performance loss in horizontal text detection. RRD achieves comparable performance with the state-of-the-art results on IC13. Moreover, compared with the most related work [26], RRD also performs better.

Figure 6: Visualization of some results on HRSC2016.

4.5. Results on the Ship Collection Benchmark

There are three level tasks in HRSC2016 dataset. Level 1 is to detect ship from backgrounds. Level 2 is to further give ship categories (*war craft, aircraft carrier etc.*) and Level 3 steps forward to get ship types (*car carrier, submarine etc.*). Basically, Level 1 task is enough to show our methods' superiority on multi-directional objects detection, so we compare our models with other carefully designed models on Level 1 task.

Some visualization results are displayed in Fig. 6. As we can see, RRD accurately detects ships with arbitrary orientations. Quantitative results are shown in Tab. 5. Instead of fine-tuning the model based on a pre-trained model, RRD is trained from scratch. Even so, RRD easily surpasses others with a promotion around 8.6 points in mAP reaching peak at 84.3 [30]. Thus, it is obvious that RRD is not only suitable for scene text detection, but also skilled in other oriented object detection.

Methods	mAP
Fast-RCNN [8]+SRBBS	55.7
Fast-RCNN [8]+SRBBS+RBB	69.6
Fast-RCNN [8]+SRBBS+RBB+RRoI	75.7
RRD(Ours)	**84.3**

Table 5: Experimental results on HSRC2016. SRBBS(Ship Rotated Bounding Boxes Space) means labeling of ships is rotated polygon; RBB(Rotated Bounding Boxes) extends Fast-RCNN to a method capable of regressing rotated bounding boxes; RRoI(Rotated Region of Interest) pooling layer is implemented based on original RoI pooling, which is specially designed for rotated bounding boxes. For detailed description, refer to [30]

4.6. Limitations

We observe that RRD fails to detect certain types of text. As shown in Fig. 7a, RRD fails to detect a whole bounding box for a text line with large character spacing. Another failure case is shown in Fig. 7b, where RRD incorrectly detects two vertical text lines as multiple horizontal ones.

We believe that detecting such text is beyond the current capability of RRD, also many other state-of-the-art methods. Higher-level semantic understanding and spatial analysis may be required to address this issue.

(a)　　　　　　(b)

Figure 7: Visualization of limited occasions. Green bounding boxes: outputs of RRD; Red dashed bounding boxes: missing ground truths.

5. Conclusion

We have proposed RRD, a novel text detector that perform classification and regression using rotation-insensitive and sensitive features respectively. This strategy is conceptually simple, yet its effectiveness and generality is well demonstrated on multiple datasets and tasks. RRD improves the dense prediction of text presence and offsets, which can be found in many modern text detectors. Potentially, these detectors will also benefit from the strategy we have adopted in RRD. In the future, we are interested in stronger rotation-sensitive features and rotation-invariant features to further improve oriented object detection. Also, since the principle of RRD goes beyond text detection and ship detection, we are interested in further exploiting its potentials in detecting other oriented objects such as those frequently appeared in aerial images [47].

6. Acknowledgements

References

[1] Coco-text challenge. http://rrc.cvc.uab.es/?ch=5&com=evaluation&task=1.

[2] D. Bazazian, R. Gmez, A. Nicolaou, L. Gmez, D. Karatzas, and A. D. Bagdanov. Fast: Facilitated and accurate scene text proposals through fcn guided pruning. *Pattern Recognition Letters*, 2017.

[3] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. Photoocr: Reading text in uncontrolled conditions. In *Proc. ICCV*, 2013.

[4] Y. Boureau, J. Ponce, and Y. LeCun. A theoretical analysis of feature pooling in visual recognition. In *ICML*, pages 111–118. Omnipress, 2010.

[5] M. Busta, L. Neumann, and J. Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *Proc. ICCV*, 2017.

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

[7] S. Dieleman, K. W. Willett, and J. Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *CoRR*, abs/1503.07077, 2015.

[8] R. B. Girshick. Fast R-CNN. In *Proc. ICCV*, 2015.

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

[10] D. M. Gonzalez, M. Volpi, and D. Tuia. Learning rotation invariant convolutional filters for texture classification. In *Proc. ICPR*, pages 2012–2017, 2016.

[11] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localisation in natural images. In *Proc. CVPR*, 2016.

[12] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li. Single shot text detector with regional attention. In *Proc. ICCV*, 2017.

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

[14] H. Hu, C. Zhang, Y. Luo, Y. Wang, J. Han, and E. Ding. Wordsup: Exploiting word annotations for character based text detection. In *Proc. ICCV*, 2017.

[15] W. Huang, Z. Lin, J. Yang, and J. Wang. Text localization in natural images using stroke feature transform and text co-variance descriptors. In *Proc. ICCV*, 2013.

[16] W. Huang, Y. Qiao, and X. Tang. Robust scene text detection with convolution neural network induced mser trees. In *Proc. ECCV*, 2014.

[17] L. G. i Bigorda and D. Karatzas. Textproposals: A text-specific selective search algorithm for word spotting in the wild. *Pattern Recognition*, 70:60–74, 2017.

[18] Y. B. Ian Goodfellow and A. Courville. Deep learning. Book in preparation for MIT Press, 2016.

[19] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *IJCV*, 116(1):1–20, 2016.

[20] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. K. Ghosh, A. D. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny. ICDAR 2015 competition on robust reading. In *Proc. ICDAR*, 2015.

[21] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. Almazán, and L. de las Heras. ICDAR 2013 robust reading competition. In *Proc. ICDAR*, 2013.

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[23] K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *CVPR*, pages 991–999. IEEE Computer Society, 2015.

[24] H. Li, P. Wang, and C. Shen. Towards end-to-end text spotting with convolutional recurrent neural networks. In *Proc. ICCV*, 2017.

[25] M. Liao, B. Shi, and X. Bai. Textboxes++: A single-shot oriented scene text detector. *CoRR*, abs/1801.02765, 2018.

[26] 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*, 2017.

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

[28] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. E. Reed. SSD: single shot multibox detector. In *Proc. ECCV*, 2016.

[29] Y. Liu and L. Jin. Deep matching prior network: Toward tighter multi-oriented text detection. In *Proc. CVPR*, 2017.

[30] Z. Liu, J. Hu, L. Weng, and Y. Yang. Rotated region based CNN for ship detection. In *Proc. ICIP*, pages 900–904, 2017.

[31] Z. Liu, L. Yuan, L. Weng, and Y. Yang. A high resolution optical satellite image dataset for ship recognition and some new baselines. In *Proc. ICPRAM*, 2017.

[32] 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.

[33] H. Rajput, T. Som, and S. Kar. An automated vehicle license plate recognition system. *IEEE Computer*, 48(8):56–61, 2015.

[34] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proc. CVPR*, 2016.

[35] J. Redmon and A. Farhadi. YOLO9000: better, faster, stronger. *CoRR*, abs/1612.08242, 2016.

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

[37] X. Rong, C. Yi, and Y. Tian. Recognizing text-based traffic guide panels with cascaded localization network. In *Proc. ECCV workshop*, 2016.

[38] B. Shi, X. Bai, and S. J. Belongie. Detecting oriented text in natural images by linking segments. In *Proc. CVPR*, 2017.

[39] B. Shi, C. Yao, M. Liao, M. Yang, P. Xu, L. Cui, S. J. Belongie, S. Lu, and X. Bai. ICDAR2017 competition on reading chinese text in the wild (RCTW-17). 2017.

[40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[41] Y. Tang and X. Wu. Scene text detection and segmentation based on cascaded convolution neural networks. *IEEE Trans. Image Processing*, 26(3):1509–1520, 2017.

[42] S. Tian, S. Lu, and C. Li. Wetext: Scene text detection under weak supervision. In *Proc. ICCV*, 2017.

[43] S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. L. Tan. Text flow: A unified text detection system in natural scene images. In *ICCV*, 2015.

[44] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao. Detecting text in natural image with connectionist text proposal network. In *Proc. ECCV*, 2016.

9

[45] A. Veit, T. Matera, L. Neumann, J. Matas, and S. J. Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *CoRR*, abs/1601.07140, 2016.

[46] Y. Wu and P. Natarajan. Self-organized text detection with minimal post-processing via border learning. In *Proc. ICCV*, 2017.

[47] G. Xia, X. Bai, J. Ding, Z. Zhu, S. J. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang. DOTA: A large-scale dataset for object detection in aerial images. In *Proc. CVPR*, 2018.

[48] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu. Detecting texts of arbitrary orientations in natural images. In *Proc. CVPR*, 2012.

[49] Z. Zhang, W. Shen, C. Yao, and X. Bai. Symmetry-based text line detection in natural scenes. In *Proc. CVPR*, 2015.

[50] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-oriented text detection with fully convolutional networks. In *Proc. CVPR*, 2016.

[51] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. EAST: an efficient and accurate scene text detector. In *Proc. CVPR*, 2017.

[52] Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao. Oriented response networks. In *CVPR*, 2017.