2018 24th International Conference on Pattern Recognition (ICPR)
Beijing, China, August 20-24, 2018

R^2CNN: Rotational Region CNN for Arbitrarily-Oriented Scene Text Detection

Yingying Jiang, Xiangyu Zhu, Xiaobing Wang, Shuli Yang, Wei Li, Hua Wang, Pei Fu and Zhenbo Luo
Samsung R&D Institute China - Beijing
{yy.jiang, xiangyu.zhu, x0106.wang, shuli.yang, wei2016.li, hua00.wang, pei.fu, zb.luo}@samsung.com

***Abstract*— Scene text detection is challenging as the input may have different orientations, sizes, font styles, lighting conditions, perspective distortions and languages. This paper addresses the problem by designing a Rotational Region CNN (R^2CNN). R^2CNN includes a Text Region Proposal Network (Text-RPN) to estimate approximate text regions and a multi-task refinement network to get the precise inclined box. Our work has the following features. First, we use a novel multi-task regression method to support arbitrarily-oriented scene text detection. Second, we introduce multiple ROIPoolings to address the scene text detection problem for the first time. Third, we use an inclined Non-Maximum Suppression (NMS) to post-process the detection candidates. Experiments show that our method outperforms the state-of-the-art on standard benchmarks: ICDAR 2013, ICDAR 2015, COCO-Text and MSRA-TD500.**

Keywords*— scene text detection; rotational region CNN; multi-task learning

I. INTRODUCTION

Text is the most fundamental tool for preserving and communicating information. It appears everywhere in daily life: on street nameplates, store signs, product packages, restaurant menus, etc. Such texts in natural environment are known as scene texts. Automatically detecting and recognizing scene texts can be very rewarding, with innumerable applications, such as in AR shopping, robots, smart cars and education. However, scene text detection is not yet to be widely used because of low detection and recognition accuracy.

Scene text detection is a challenging problem because scene texts have different orientations, sizes, font styles, lighting conditions, perspective distortions and languages. The traditional sliding-window or Connected Components (CCs) based methods[4~12] are weak at dealing with different font styles, lighting conditions, perspective distortions and orientations. In 2013, the traditional method on focused scene text detection only achieved an F-measure of 0.759 on ICDAR 2013 dataset [1]. Recent deep learning based object detection and scene text detection methods [13~30,36,37,38] achieved better results, but the arbitrarily-oriented scene text detection problem is not yet well addressed. For example, Faster R-CNN [13] is extensively used for general object detection, but it cannot detect scene texts with non-horizontal orientations. EAST [14] is able to detect oriented scene texts, but does not perform well on vertical texts. In this paper we focus on arbitrarily-oriented scene text detection.

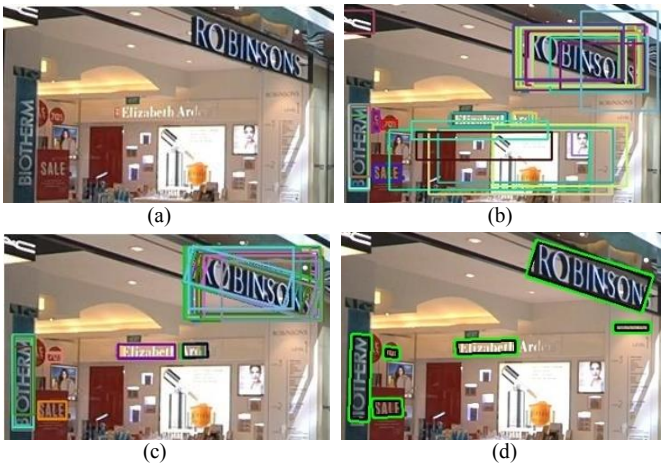

Fig. 1. The procedure of the proposed method R^2CNN. (a) Original input image; (b) text regions (axis-aligned bounding boxes) generated by Text Region Proposal Network (Text-RPN); (c) predicted axis-aligned boxes and inclined minimum area boxes (each inclined box is associated with an axis-aligned box, and the associated box pair is indicated by the same color); (d) detection result after inclined Non-Maximum Suppression (NMS).

Rotational Region CNN (R^2CNN) introduced in this paper is a scene text detection method to meet the challenges of arbitrarily-oriented scene text detection. To capture the complex information presented by scene texts in the wild, it applies a coarse-to-fine strategy, inspired by Faster R-CNN [13]. It first gets approximate axis-aligned text regions and then obtains refined axis-aligned box and inclined box. R^2CNN (Figure 1 and Figure 3) consists of three steps: a Text Region Proposal Network (Text-RPN) to propose text regions (Figure 1(b)), a multi-task refinement network to predict axis-aligned boxes and inclined boxes (Figure 1(c)), and an inclined Non-Maximum Suppression (NMS) step to get the final detection results (Figure 1(d)).

The method effectively detects scene texts of different sizes, different orientations and different languages at word and sentence levels. It produces state-of-the-art results on ICDAR 2013 [1], ICDAR 2015 [2], COCO-Text [3] and MSRA-TD500 [4].

The contributions of this paper are as follows:

- A new text detection framework R^2CNN for arbitrarily-oriented scene texts (Figure 3) is proposed. It consists of three stages: a Text-RPN to propose text regions, a multi-task

refinement network to predict inclined box, and an inclined NMS stage.

- Arbitrarily-oriented scene text detection is formulated as a multi-task problem. R^2CNN predicts both the axis-aligned box and inclined box for each text region generated by the Text-RPN for the first time.

- To make the most of text characteristics, we do several ROIPoolings with different pooled sizes (7×7, 11×3, 3×11) for each RPN proposal. The concatenated features are then used for further detection.

- An inclined NMS is designed to post-process the detection candidates to get the final results.

II. Related Work

Traditional sliding-window-based and Connected-components(CC)-based scene text detection methods [4~12] had been widely used before deep neural networks became the most promising machine learning tool. Sliding-window based methods move a multi-scale window over an image and classify the current patch as character or non-character. CC-based approaches, particularly the Maximally Stable Extremal Regions (MSER) methods, get character candidates by extracting CCs. MSER achieved good performances in ICDAR 2013 [1] and ICDAR 2015 [2]. These traditional methods adopt a bottom-up strategy and often need several steps to detect texts (e.g., character detection, text line construction and text line classification).

General object detection is a popular research field. Deep-learning-based techniques have advanced object detection substantially. One family of object detection methods relies on region proposal, such as R-CNN [21], SPPnet [19], Fast R-CNN [22], Faster R-CNN [13], R-FCN[20] and Mask R-CNN [35]. Another family of object detectors directly estimates object candidates, such as SSD [23] and YOLO [24]. While these methods are effective for general object detection, they cannot detect oriented objects and thus cannot be used for arbitrarily-oriented scene text detection.

Recently, deep-learning-based scene text detection methods have become an attractive topic. TextBoxes [26], DeepText [25], FCRN [28] and FEN[38] are designed to generate axis-aligned detection boxes and do not address the text orientation problem. CTPN [27] can detect oriented scene texts, but it is not suitable for highly inclined texts. To detect arbitrarily-oriented scene texts, FCN-based method [29], RRPN [18], SegLink [15], EAST [14], deep direct regression [17], Multi-scale FCN [30], and PixelLink[36] are proposed. FCN-based method [29] consists of three steps: text block detection by text block FCN, multi-oriented text line candidate generation based on MSER and text line candidates classification. RRPN [18] is based on Faster R-CNN and utilizes a modified RPN to generate inclined proposals and carries out the following classification and regression using the inclined proposals. SegLink [15] detects oriented texts by detecting segments and links. It works well on text lines with arbitrary lengths. EAST [14] is designed to yield fast and accurate text detection in natural scenes, but vertical texts are not well detected. DMPNet [16] is designed to detect text with tighter quadrangle. Deep direct regression [17] is proposed for multi-oriented scene text detection. Multi-scale FCN [30] is used together with cascaded instance aware segmentation to detect arbitrarily oriented words in the wild. These methods are able to detect oriented scene texts, but their performances are not good enough due to the challenges involved.

Our method is inspired by the general object detection framework Faster R-CNN [13]. We design Text-RPN to generate text regions and then predict the orientation information based on the proposed text regions.

III. Methodology

In this section, we introduce our approach to detect arbitrarily-oriented scene texts. Figure 3 shows the architecture of the proposed Rotational Region CNN (R^2CNN). We first present how we formalize the arbitrarily-oriented text detection problem and then introduce the details of R^2CNN. After that, we describe our training objectives.

A. Problem Definition

For arbitrarily oriented scene texts, the axis-aligned (horizontal) bounding box is not enough to describe the text area accurately. Thus, in this paper, we try to detect arbitrarily-oriented text with both axis-aligned and inclined bounding boxes.

Figure 2 shows the detection targets of our approach. In Figure 2(a), we estimate the inclined bounding box by predicting the coordinates of the first two points in clockwise and the height of the bounding box (x1,y1,x2,y2,h). Figure 2 (b) shows that we estimate the axis-aligned box by predicting the center point coordinates and the box's width and height (cx,cy,w,h).

For inclined bounding box estimation, the first point (x1,y1) always means the point at the left-top corner of the scene text (the solid red point in Figure 2(a)). Although it is straightforward to represent an inclined box by using an angle to represent its orientation, we don't adopt this strategy because the angle target is not stable in some special points. For example, a rectangle with rotation angle 90^0 is very similar to the same rectangle with rotation angle -90^0, but their angles are quite different. This makes the network hard to learn to detect vertical texts.

(a) (b)

Fig. 2. Detection targets of arbitrarily-oriented scene text detection. a) The inclined minimum area rectangle; b) the axis-aligned box of the scene text.

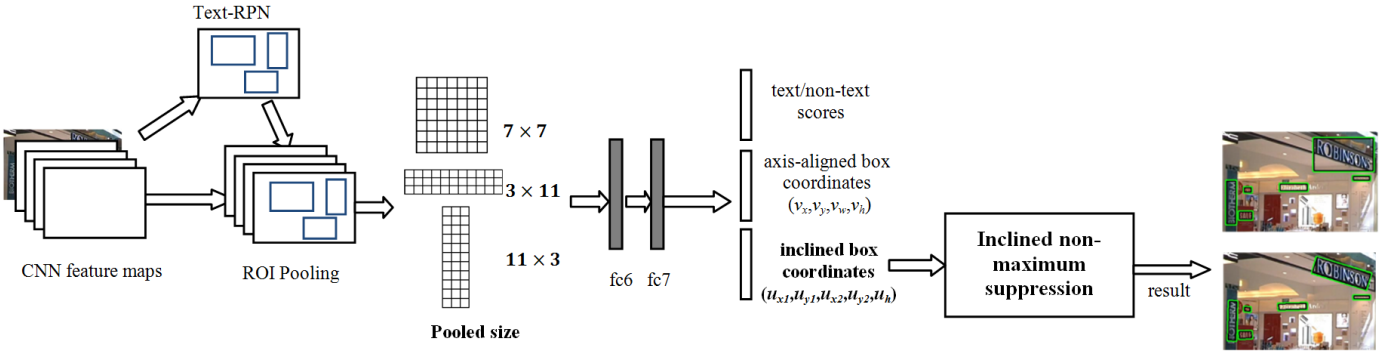

Fig. 3. Architecture of Rotational Region CNN (R^2CNN). The feature extraction is based on VGG16. The Text-RPN is used for proposing axis-aligned bounding boxes that enclose the arbitrarily-oriented texts. For each box generated by Text-RPN, three ROIPoolings with different pooled sizes are performed and the pooled features are concatenated for predicting the text scores, axis-aligned box (v_x,v_y,v_w,v_h) and inclined minimum area box (u_{x1},u_{y1},u_{x2},u_{y2},u_h). Then an inclined non-maximum suppression is conducted on the inclined boxes to get the final result.

B. *Network Design*

Overview of R^2CNN. To address the challenge of arbitrarily-oriented scene text detection and capture the complex information presented by scene texts in the wild, R^2CNN adopts a coarse-to-fine process. The first stage uses simple features to estimate approximate text regions. The second stage uses multiple ROIPooling features to get the precise inclined text area based on the text regions generated in the first stage. Finally, an inclined NMS step is used to remove redundant text boxes.

Figure 3 shows the architecture of R^2CNN. R^2CNN extracts features using VGG16. Text-RPN generates text region proposals, which are axis-aligned bounding boxes that enclose arbitrarily-oriented texts (Figure 1(b)). For each proposal, we perform ROIPoolings with different pooled sizes (7×7, 11×3, 3×11). The pooled features are concatenated for predicting text/non-text scores, axis-aligned boxes and inclined minimum area boxes (Figure 1(c)). After that, the inclined boxes are post-processed by inclined NMS to get the detection results (Figure 1(d)). The output includes both the inclined box and its corresponding axis-aligned box.

Text-RPN for proposing axis-aligned boxes. The Text-RPN is designed to generate axis-aligned bounding boxes that enclose arbitrarily-oriented texts. The text in the axis-aligned box belongs to one of the following situations: a) the text is in the horizontal direction; b) the text is in the vertical direction; c) the text is in a diagonal direction of the axis-aligned box. Figure 1(b) shows an example where the Text-RPN successfully generates axis-aligned boxes for arbitrarily-oriented texts.

Compared to general objects, there are more small scene texts. We design smaller anchors in Text-RPN to deal with small scene texts. Our experiments also confirm that this design is effective.

ROIPoolings of different pooled sizes. As the widths of some texts are much larger than their heights, we try to use three ROIPoolings with different sizes to catch more text characteristics of each text candidate proposed by Text-RPN. The pooled features are concatenated for further detection.

Specifically, we use three pooled sizes: 7×7, 11×3 and 3×11. The pooled size 7×7 is supposed to catch generic text features for detection. The pooled size 3×11 is supposed to catch more horizontal features and help the detection of the horizontal text whose width is much larger than its height. The pooled size 11×3 is supposed to catch more vertical features and be useful for vertical text detection that the height is much larger than the width.

Multi-task regression for text/non-text scores, axis-aligned boxes, and inclined minimum area boxes. Multi-task learning is effective in improving performance of object detection and semantic segmentation [35]. We utilize multi-task regression in scene text detection to improve the performance of arbitrarily oriented scene text detection.

Specifically, the concatenated features from several ROIPoolings are shared by downstream tasks of text/non-text classification as well as axis-aligned and inclined bounding box regressions of text regions. Each inclined box is associated with an axis-aligned box (Figure 1(c), 2 and 4(a)).

Inclined NMS. Since NMS may miss to detect inclined scene texts, R^2CNN designs inclined NMS for arbitrarily-oriented scene texts (Figure 4).

Non-Maximum Suppression (NMS) is extensively used to post-process detection candidates by current object detection methods. As we estimate both axis-aligned and inclined bounding boxes, we can do either normal NMS or inclined NMS. In inclined NMS, the calculation of Intersection-over-Union (IoU) is modified to work with inclined bounding boxes [18]. We use inclined NMS in R^2CNN, as it performs better on inclined candidates (Figure 4).

Figure 4 compares the two NMS methods. Figure 4(a) shows predicted candidate boxes, each axis-aligned bounding box associated with an inclined one. Figure 4(b) shows the result of normal NMS and Figure 4(c) shows that of inclined NMS. We see that the normal NMS erroneously suppressed the text in red dashed box, as its axis-aligned bounding box has a high IoU with another (Figure 4(d)).

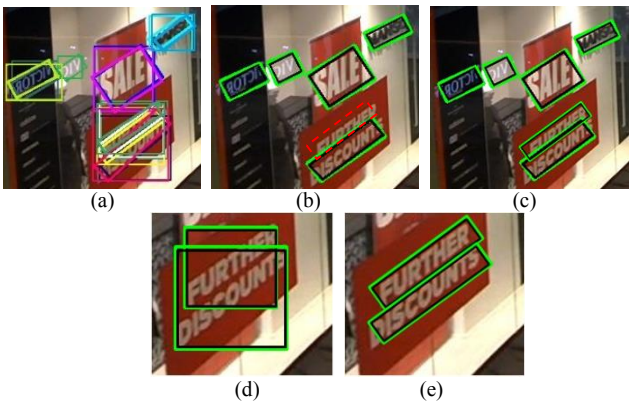

Fig. 4. Inclined NMS. (a) The candidate axis-aligned boxes and inclined boxes; (b) the detection results based on normal NMS on axis-aligned boxes (the green boxes are the correct detections, and the red dashed box is the box that is not detected); (c) the detection results based on inclined NMS on inclined boxes; (d) an example of two axis-aligned boxes; (e) an example of two inclined boxes.

C. Training Objective (Multi-task loss)

The training loss on Text-RPN is the same as Faster R-CNN [13]. In this section, we focus on the loss function of R^{2}CNN on each axis-aligned box proposal generated by Text-RPN.

Our loss function defined on each proposal is the sum of a text/non-text classification loss and a box regression loss. The box regression loss consists of two parts: the loss of axis-aligned boxes and the loss of inclined minimum area boxes. The multi-task loss function on each proposal is defined as:

$$\begin{aligned} L(p,t,v,v^*,u,u^*) = & L_{\text{cls}}(p,t) \\ & +\lambda_1 t \sum_{i\in\{x,y,w,h\}} L_{reg}(v_i,v_i^*) \\ & +\lambda_2 t \sum_{i\in\{x1,y1,x2,y2,h\}} L_{reg}(u_i,u_i^*) \end{aligned} \quad (1)$$

λ_1 and λ_2 are balancing parameters that control the trade-off between the three terms.

The box regression only conducts on text. t is the indicator of the class label. Text is labeled as 1 ($t = 1$), and background is labeled as 0 ($t = 0$). The parameter $p = (p_0, p_1)$ is the probability over text and background classes computed by the softmax. $L_{\text{cls}}(p,t) = -\log p_t$ is the log loss for true class t.

$v = (v_x, v_y, v_w, v_h)$ is a tuple of true axis-aligned bounding box regression targets including coordinates of the center point and its width and height, and $v^* = (v_x^*, v_y^*, v_w^*, v_h^*)$ is the predicted tuple for the text label. $u = (u_{x1}, u_{y1}, u_{x2}, u_{y2}, u_h)$ is a tuple of true inclined bounding box regression targets including coordinates of first two points of the inclined box and its height, and $u^* = (u_{x1}^*, u_{y1}^*, u_{x2}^*, u_{y2}^*, u_h^*)$ is the predicted tuple for the text label. We use the parameterization for v and v^* given in [21], in which v and v^* specify scale-invariant translation and log-space height/width shift relative to an object proposal. For inclined bounding boxes, the parameterization of (u_{x1}, u_{y1}), (u_{x2}, u_{y2}), (u_{x1}^*, u_{y1}^*) and (u_{x2}^*, u_{y2}^*) is the same as that of (v_x, v_y); the parameterization of u_h and u_h^* is the same as that of v_h and v_h^*.

Let (w, w^*) indicates (v_i, v_i^*) or (u_i, u_i^*), $L_{reg}(w, w^*)$ is defined as the smooth L1 loss as in [13]:

$$L_{reg}(w,w^*) = \text{smooth}_{\text{L1}}(w - w^*) \quad (2)$$

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (3)$$

IV. Experiments

We evaluate our method on the following standard benchmarks: ICDAR 2013[1], ICDAR 2015[2], COCO-Text [3] and MSRA-TD500 [4].

A. Benchmark Datasets

ICDAR 2013 is used in task 2.1 Focused Scene Text Detection of ICDAR 2015 Robust Reading Competition. It includes 233 test images. The scene texts are horizontal and labeled by axis-aligned boxes at word level.

ICDAR 2015 is used in task 4.1 Incidental Scene Text Detection of ICDAR 2015 Robust Reading Competition. It includes 1000 training images and 500 testing images. The scene texts have different orientations and are labeled by inclined boxes at word level.

COCO-Text includes 43,686 images for training and 20,000 images for testing with scene text from MS-COCO (Microsoft COCO) dataset. They are labeled by axis-aligned boxes.

MSRA-TD500 contains 200 test images that contain arbitrarily-oriented texts in both Chinese and English. The texts are labeled by inclined boxes at sentence level.

B. Implementation Details

Training dataset. We train one model for ICDAR2013 and ICDAR2015 using the training datasets of both. To support arbitrarily-oriented scene text detection, we augment the data by rotating images at the following angles (-90, -75, -60, -45, -30, -15, 0, 15, 30, 45, 60, 75, 90).

For MSRA-TD500, we utilize training data in RCTW-17 [32] to train the model. It contains 12,000 training images that include Chinese and English. Each label corresponds to a word or sentence.

For COCO-Text, we utilize the COCO-Text training set to train the model.

Training. Our network is initialized with pre-trained VGG16 model for ImageNet classification [33]. Training is performed end-to-end. All models are trained 20×10^4iterations in total. Learning rates start from 10^{-3}, and are multiplied by $\frac{1}{10}$ after 5×10^4, 10×10^4 and 15×10^4 iterations. We use 0.0005 weight decay and 0.9 momentum. Multi-scale training is adopted.

TABLE I. RESULTS OF R^{2}CNN UNDER DIFFERENT SETTINGS ON ICDAR 2015

Approaches	Anchor scales	Axis-aligned box (λ_1) and inclined box (λ_2)	Pooled sizes	Inclined NMS	Recall	Precision	F-measure	Time
Faster R-CNN	(8,16,32)	$\lambda_1 = 1, \lambda_2 = 0$	7×7		0.591	0.543	0.566	0.38s
R^{2}CNN-1	(8,16,32)	$\lambda_1 = 0, \lambda_2 = 1$	7×7		0.636	0.612	0.624	0.39s
R^{2}CNN-2	(8,16,32)	$\lambda_1 = 1, \lambda_2 = 1$	7×7		0.682	0.688	0.685	0.4s
R^{2}CNN-3	(4, 8,16,32)	$\lambda_1 = 1, \lambda_2 = 1$	7×7		0.727	0.732	0.729	0.41s
R^{2}CNN-4	(4, 8,16,32)	$\lambda_1 = 1, \lambda_2 = 1$	7×7, 11×3, 3×11		0.747	0.741	0.744	0.45s
				Y	0.743	0.764	0.753	0.45s

C. Quantitative Results

Effectiveness of our design. First, we conduct several experiments to confirm the effectiveness of our design. Table I summarizes the results of our models under different settings on ICDAR 2015. They are tested at 720 pixel. We compare the following models: Faster R-CNN[13], R^{2}CNN-1, R^{2}CNN-2, R^{2}CNN-3 and R^{2}CNN-4. We mainly focus on evaluating the influence of the axis-aligned box regression (λ_1) and the inclined box regression (λ_2), the effect of multiple ROIPoolings, and the impact of the anchor scales and NMS strategy. All models are trained on the same ICDAR 2013 and ICDAR 2015 training datasets introduced in the last section.

We can see that all our designs are better than the Faster R-CNN baseline. R^{2}CNN-2 improves over R^{2}CNN-1 by 0.06 in F-measure, showing that the multi-task design is effective. R^{2}CNN-3 gains another 0.044 improvement, which indicates that smaller anchor scales are critical. R^{2}CNN-4 shows multiple ROIPoolings and inclined NMS are helpful. They bring 0.024 improvement in F-measure.

Performance on oriented scene texts. We perform multi-scale testing on R^{2}CNN. The results on oriented scene text datasets can be seen in Table II, III and IV.

Table II shows the results on ICDAR 2015[2]. We can see that R^{2}CNN can work well on incidental scene text detection and get F-measure of 0.839. Table III shows the results on COCO-Text[3]. R^{2}CNN can detect complex everyday scene texts well. And it can get F-measure of 0.615, which is significantly better than other methods. Table IV is the results of R^{2}CNN on MSRA-TD500 [4]. R^{2}CNN can handle mixed language well. It can achieve F-measure of 0.852.

Performance on horizontal scene texts. Table V shows the results of R^{2}CNN on ICDAR2013 focused horizontal scene texts. It shows that our R^{2}CNN performs better than all other evaluated methods for horizontal scene texts.

Test time. The test times in Table I are measured on single Tesla K80 GPU. Under the same resolution, our method only marginally increases detection time compared to the Faster R-CNN baseline.

D. Qualitative Results

Figure 5 illustrates qualitative results on ICDAR2013, ICDAR2015, COCO-Text and MSRA-TD500 by R^{2}CNN. It shows that our method can deal with scene texts with arbitrary orientations, different languages, non-uniform illuminations and different text lengths at word level or sentence level.

TABLE II. RESULTS ON ICDAR2015 INCIDENTAL SCENE TEXT

Approaches	Recall	Precision	F-measure
R^{2}CNN	**0.829**	0.850	**0.839**
PixelLink[36]	0.820	0.855	0.837
TextBoxes++_ MS [37]	0.785	**0.878**	0.829
He et al.[17]	0.800	0.820	0.810
EAST[14]	0.783	0.833	0.807
RRPN*[18]	0.770	0.840	0.800
SegLink[15]	0.768	0.731	0.750
DMPNet[16]	0.682	0.732	0.706
Yao et al. [34]	0.587	0.723	0.648
He et al. [30]	0.540	0.760	0.630
CTPN[27]	0.520	0.740	0.610

TABLE III. RESULTS ON COCO-TEXT

Approaches	Recall	Precision	F-measure
R^{2}CNN	**0.639**	0.593	**0.615**
TextBoxes++_ MS [37]	0.567	**0.609**	0.587
EAST[14]	0.324	0.504	0.395
Veit et al. [3]	0.233	0.838	0.365
Yao et al.[34]	0.271	0.432	0.333

TABLE IV. RESULTS ON MSRA-TD500

Approaches	Recall	Precision	F-measure
R^{2}CNN	**0.827**	**0.879**	**0.852**
PixelLink[36]	0.732	0.830	0.778
SegLink[15]	0.700	0.860	0.770
EAST[14]	0.674	0.873	0.761
Yao et al. [34]	0.753	0.765	0.759
RRPN*[18]	0.690	0.820	0.750
Zhang et al. [29]	0.670	0.830	0.740
Heet al. [17]	0.700	0.770	0.740
Yin et al. [10]	0.630	0.810	0.740

TABLE V. RESULTS ON ICDAR2013 FOCUSED SCENE TEXT

Approaches	Recall	Precision	F-measure
R^{2}CNN	**0.898**	0.937	**0.917**
FEN_MT[38]	0.893	0.941	0.916
RRPN*[18]	0.879	**0.949**	0.913
CTPN [27]	0.830	0.930	0.880
He et al. [17]	0.810	0.920	0.860
SegLink[15]	0.830	0.877	0.853
He et al. [30]	0.790	0.930	0.850
TextBoxes[26]	0.830	0.880	0.850
DeepText [25]	0.830	0.870	0.850

(a) ICDAR 2013: detection on focused horizontal scene texts at word level

(b) ICDAR2015: detection on incidental arbitrarily-oriented scene texts at word level

(c) COCO-Text: detection on complex everyday scene texts at word level

(d) MSRA-TD500: detection of Chinese and English at line level

Fig. 5. Detection results on different benchmarks. The green boxes are the correct detection results. The red boxes are false positives. The dashed red boxes are false negatives.

V. CONCLUSION

This paper introduces Rotational Region CNN (R^2CNN) for detecting arbitrarily oriented scene texts. It is inspired by Faster R-CNN [13]. It first uses Text-RPN to propose text regions, then utilizes multiple ROIPooling features to predict both axis-aligned and inclined bounding boxes, and finally uses inclined NMS for post-processing. R^2CNN can detect scene texts of different orientations and languages at word/sentence level. It achieves state-of-art results on ICDAR2013 [1], ICDAR2015 [2], COCO-Text [3] and MSRA-TD500 [4].

In the future, we will extend our work in the following aspects. First, we will optimize our network in terms of time and spatial efficiency to make it able to be applied on embedded devices. Second, we will investigate detection and recognition tasks in a holistic end-to-end network.

[1] D. Karatzas, F. Shafait, S. Uchida, et al. ICDAR 2013 Robust Reading Competition. ICDAR 2013.

[2] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, et al. ICDAR 2015 Competition on Robust Reading. ICDAR 2015.

[3] A. Veit, T. Matera, et al. COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images. arXiv:1601.07140, 2016.

[4] C. Yao, X. Bai, et al. Detecting Texts of Arbitrary Orientations in Natural Images. CVPR 2012.

[5] L. Neumann and J. Matas. Scene Text Localization and Recognition with Oriented Stroke Detection. ICCV 2013.

[6] X. C. Yin, X. Yin, K. Huang, and H. Hao. Robust Text Detection in Natural Scene Images. IEEE Trans. on PAMI,36(5):970–983, 2014.

[7] W. Huang, Z. Lin, J. Yang, and J. Wang. Text Localization in Natural Images Using Stroke Feature Transform and Text CovarianceDescriptors. ICCV 2013.

[8] 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. CVPR 2015.

[9] L. Sun, Q. Huo, and W. Jia. A Robust Approach for Text Detection from Natural Scene Images. Pattern Recognition, 48(9):2906–2920, 2015.

[10] X. Yin, W. Pei, J. Zhang, and H. Hao. Multi-OrientationScene Text Detection with Adaptive Clustering. PAMI,37(9):1930–1937, 2015.

[11] L. Kang, Y. Li, and D. Doermann. Orientation Robust Text Line Detection in Natural Images. CVPR 2014.

[12] H. Cho, M. Sung, and B. Jun. Canny Text Detector: Fast and Robust Scene Text Localization Algorithm. CVPR 2016.

[13] S. Ren, K. He, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. NIPS 2015.

[14] X. Zhou, C. Yao, et al. EAST: An Efficient and Accurate Scene Text Detector. CVPR 2017.

[15] B. Shi, X. Bai, and S, Belongie. Detecting Oriented Text in Natural Images by Linking Segments. CVPR 2017.

[16] Y. Liu, and L, Jin. Deep Matching Prior Network: Toward Tighter Multi-oriented Text Detection. CVPR 2017.

[17] W. He, X.Y. Zhang, et al. Deep Direct Regression for Multi-Oriented Scene Text Detection. ICCV 2017.

[18] J. Ma, W. Shao, et al. Arbitrary-Oriented Scene Text Detection via Rotation Proposals. arXiv:1703.01086, 2017.

[19] K. He, X. Zhang, S. Ren, and J. Sun. SPPNet: Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. ECCV 2014.

[20] J. Dai, Y. Li, K. He, and J. Sun. R-FCN: Object Detection via Region-based Fully Convolutional Networks. NIPS 2016.

[21] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR 2014.

[22] R. Girshick. Fast R-CNN. ICCV 2015.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed. SSD: Single Shot MultiBox Detector. ECCV 2016

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection. CVPR 2016.

[25] Z. Zhong, L. Jin, S. Zhang, and Z. Feng. Deeptext: A Unified Frameworkfor Text Proposal Generation and Text Detection in Natural Images. arXiv:1605.07314, 2016.

[26] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu. TextBoxes: A Fast Text Detector with a Single Deep Neural Network. AAAI 2017.

[27] Z. Tian, W, Huang, T. He, P. He, and Y. Qiao. Detecting Text in Natural Image with Connectionist Text Proposal Network. ECCV 2016.

[28] A.Gupta, A.Vedaldi, and A.Zisserman. Synthetic Data for Text Localisation in Natural Images. CVPR2016.

[29] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-Oriented Text Detection with Fully Convolutional Networks. CVPR 2016

[30] D. He, X. Yang, C. Liang, Z. Zhou, A.G. Ororbi II, et al. Multi-Scale FCN with Cascaded Instance Aware Segmentation for Arbitrary Oriented Word Spotting in the Wild. CVPR 2017.

[31] L.C. Chen, G. Papandreou, et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. arXiv: 1606.00915

[32] RCTW-17: http://mclab.eic.hust.edu.cn/icdar2017chinese/

[33] K.Simonyan, and A.Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. NIPS2015.

[34] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao. Scene Text Detection via Holistic, Multi-Channel Prediction. arXiv:1606.09002, 2016.

[35] K. He, G. Gkioxari, P. Dollar, and Ross. Girshick. Mask R-CNN. ICCV 2017.

[36] D. Deng, H. Liu, X. Li, D. Cai. PixelLink: Detecting Scene Text via Instance Segmentation. AAAI 2018.

[37] M. Liao, B. Shi, X. Bai. TextBoxes++: A Single-Shot Oriented Scene Text Detector. arXiv: 1801.02765v1

[38] S. Zhang, Y. Liu, L. Jin, C. Luo. Feature Enhancement Network: A Refined Scene Text Detector. AAAI 2018.