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

Scene Text Detection with Recurrent Instance Segmentation

Wei Feng[1,2], Wen-Hao He[1,2], Fei Yin[1,2], Cheng-Lin Liu[1,2,3]
[1]National Laboratory of Pattern Recognition, Institutaion of Automation of Chinese Academy of Sciences
95 Zhongguancun East Road, Beijing 100190, P.R. China
[2]University of Chinese Academy of Sciences, Beijing, P.R. China
[3]CAS Center for Excellence of Brain Science and Intelligence Technology, Beijing, P.R. China
Email: {wei.feng, wenhao.he, fyin, liucl}@nlpr.ia.ac.cn

***Abstract*—Convolutional Neural Network (CNN) based scene text detection methods mostly employ the semantic segmentation (text/non-text classification) task to localize the regions of texts. However, they cannot distinguish different text-lines like instance segmentation. In this paper, we propose a novel framework based on Fully Convolutional Networks (FCN) and Recurrent Neural Network (RNN) to achieve both scene text detection and instance segmentation. The FCN is used to classify text and non-text regions, and the RNN utilizes the features extracted by FCN to simultaneously detect and segment one text instance at each time step. Meanwhile, it also extracts bounding boxes by a much simpler way than the non-maximum suppression (NMS) method. The proposed method achieves competitive results on two public benchmarks including ICDAR 2015 Incidental Scene Text Dataset and ICDAR 2013 Focused Scene Text Dataset. Moreover, the benefits of adding regression task in the RNN module are manifested.**

I. Introduction

Scene text detection is a challenging problem due to the cluttering of background, as well as the variation of illumination and perspective of camera. Current state-of-the-art results are almost achieved by Convolutional Neural Network (CNN) based methods, which contain two main subtasks. First, a segmentation task is used to classify text/non-text regions. Second, a regression task is used to determine the bounding boxes. Segmentation task plays an important role in scene text detection as it is the basis of other tasks in either an explicit or implicit way. In [1] [2], the segmentation task localizes text regions explicitly as it directly predicts the text score map. While in methods [3] [4] based on the Regional Proposal Network (RPN) in [5], the segmentation task plays implicitly since text regions that are similar to anchor shape priors are highlighted. However, when text-lines lie close to each other and features are down-sampled, semantic segmentation usually suffers from adhesion problem as shown in Fig 1.(b).

Recently, instance segmentation [6] [7] [8] has proposed to combine semantic segmentation and object detection together, and can localize objects at pixel-level as shown in Fig 1.(c). Unlike segmentation, instance segmentation can distinguish different instances of the same category, and unlike detection, it can localize the object at pixel-level rather than roughly gives a bounding box. To take advantage of multi-task learning, some methods are proposed to learn detection and instance segmentation simultaneously. For example, He *et al.* [9] propose Mask-RCNN which is extended from Faster R-CNN by adding a branch for predicting segmentation masks in parallel with the existing branches for bounding box regression and classification. However, these methods may not be effective for scene text detection and instance segmentation. The reasons are mainly in two folds. First, most existing methods for instance segmentation rely on object proposals, and then segmentation is conducted within each proposal. As a consequence, instance segmentation is sensitive to the quality of proposals. In other words, if the proposal generation task fails, the instance segmentation task would not be proceeded. Second, anchor mechanism in Faster-RCNN may not be able to generate suitable text proposals as multi-oriented scene texts could be long and heavily inclined as illustrated in [1]. Based on the analysis above, methods for detecting and segmenting text instances simultaneously with no proposals could be a better choice. Recent works like those in [10] [11] adopt Recurrent Neural Network (RNN) to realize instance-level detection or segmentation at each time step with no bounding box proposals. These works provide a new clue to deal with the problem of scene text detection.

(a) (b) (c)

Fig. 1. Semantic segmentation usually suffers from adhesion problem. (a): original image. (b): ground truth for semantic segmentation. (c): ground truth for instance segmentation. Best view in color.

In this paper, we propose an end-to-end trainable scene text detection and instance segmentation framework without region proposals. The proposed framework achieves both detection and instance segmentation using Fully Convolutional Networks (FCN) [12] and RNN. The FCN is used to classify text and non-text regions and the RNN utilizes the features extracted by FCN to detect and segment text instances iteratively, one at each time step. Meanwhile, it can extract text bounding

978-1-5386-3788-3/18/$31.00 ©2018 IEEE

boxes directly without the commonly used non-maximum suppression (NMS). Moreover, irregular quadrilateral boundaries can be predicted by adding a regression task.

Our contributions are in three folds: first, we propose a novel scene text detection framework which achieves both detection and instance segmentation. Second, this is the first RNN based method needing no proposals for text instance segmentation and detection. Third, the RNN can predict irregular quadrilateral boundaries of text instances, and this gives more accurate location of text instances.

The remainder of this paper is organized as follows: Section II reviews related works of scene text detection and instance segmentation. Section III describes the proposed method. Section IV presents experimental results on benchmarks and analyses the benefits of regression task. This paper is concluded in Section V.

II. Related Work

A. Scene Text Detection

The existing methods for scene text detection can be roughly grouped into three categories: character based, text-line based and word based methods.

Character based methods like [13] [14] first localize characters in sliding window fashion or utilizing connected components and then integrate them into words by either rule based methods or graphic models. Text-line based methods like [15] [16] detect text-lines firstly by exploiting symmetry property or salient maps and then separate each line into multiple words. These methods are prone to error accumulation and inefficiency because of the multiple stages.

Recently, some word based methods detect words directly in the similar way as generic object detection. On the basis of Faster R-CNN, Zhong *et al.* [3] propose DeepText in which the Region Proposal Network (RPN) and the RoI pooling layer are redesigned. Liao *et al.* [4] follow SSD [17] and adapt the model to text by adjusting the network parameters. Similar to Densebox [18], He *et al.* [1] use the FCN to detect multi-oriented scene text. These methods have achieved superior performance over conventional methods as they eliminate unnecessary intermediate steps. The proposed method in this paper belongs to this category.

B. Instance Segmentation

Recent deep neural network based instance segmentation methods can be roughly grouped into two categories: proposal based and proposal-free methods.

Proposal based methods firstly generate object proposals by RPN or sliding windows and then segmentation is proceeded within the proposals. Dai *et al.* [8] integrate RPN into a multi-task network cascade (MNC) for instance segmentation. Li *et al.* [19] utilize the segment proposal system to predict a set of position-sensitive output channels for fully convolutional instance segmentation. However, these methods all cascade proposal generation module and instance segmentation module, which cause errors accumulation and inefficiency.

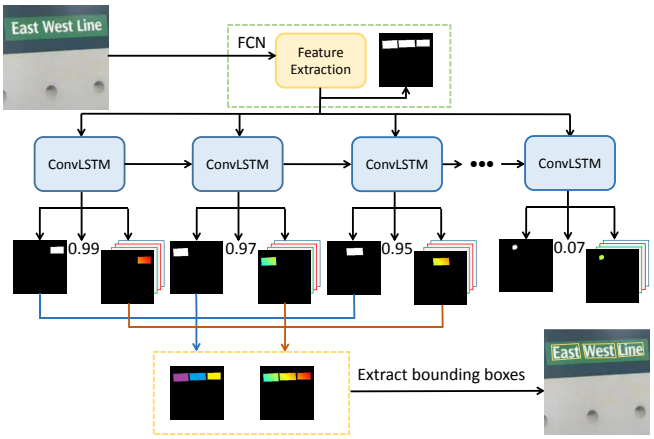

Fig. 2. The diagram of the proposed method.

Proposal-free methods attempt to achieve instance segmentation directly without proposals. Liang *et al.* [20] propose the Proposal-free Network (PFN) which predicts the number of instance, each pixel's label and its enclosing bounding box, but their result is sensitive to predicting the number of instance accurately. The method of [21] identifies the individual instances based on their depth ordering. However, it's hard to distinguish instances at roughly identical depths. Above all, these proposal-free methods cannot perform detection and segmentation at the same time. The proposed framework is also a proposal-free method. It can further detect and segment text instance-by-instance with RNN.

III. Methodology

A. Network Architecture

The proposed framework is diagrammed in Figure 2. The framework consists of two major parts: a FCN for text segmentation (text/non-text classification) and a RNN for text instance segmentation and bounding box regression. As the FCN and RNN can share features, the whole framework is end-to-end trainable.

The FCN is designed for pixel-level classification. As scene text feature is not as complicated as that of generic objects, we replace the network proposed in [12] with a simpler one which has fewer channels. Meanwhile, considering the sizes of word regions vary tremendously, we also fuse convolutional feature in multiple scales. Furthermore, we only up-sample the fused feature to quarter size of the input image to save computation. At last, we use a 1×1 convolution to project 128 channels of feature maps into 2 channels for pixel-wise classification between text and non-text. The detailed structure and parameters are diagrammed in Figure 3.

The goal of RNN is to segment and detect text instances sequentially. To avoid the vanishing gradient problem, we adopt the Long Short-term Memory (LSTM) networks in [22]. However, traditional LSTM may not work well in this problem as both the input and the output are feature maps rather than feature vectors. In remedy of this, we change the LSTM

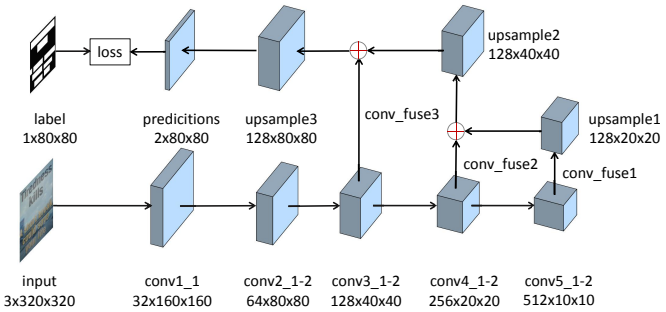

Fig. 3. The structure of the FCN.

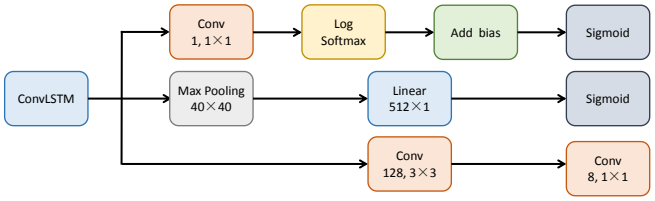

Fig. 4. The structure of the three output branches.

units to Convolutional Long Short-term Memory (ConvLSTM) units, in which fully connected layers are replaced by convolutional ones. The ConvLSTM takes the 128-channel of feature maps extracted by the FCN as input for each time step. The initial hidden state of the ConvLSTM is initialized to 0. After the first iteration, the hidden state is updated which contains the information about the former instance.

For each iteration, the RNN has three output branches. The first output is a map that indicates pixels within the text instance that should be segmented in the current iteration. The second output is 8-channel map which means the offset from coordinate of each quadrilateral vertex to each point. The third output is the estimated probability that the current segmented candidate is a text instance, which can be used as the stop condition in the test phase. The detailed structure and parameters are diagrammed in Figure 4.

B. Ground Truth and Loss Function

For end-to-end training of the framework, the whole loss function can be formulated as

$$\mathcal{L} = \mathcal{L}_{seg} + \lambda_{ins}\mathcal{L}_{ins} + \lambda_{reg}\mathcal{L}_{reg} + \lambda_{stop}\mathcal{L}_{stop}, \quad (1)$$

where $\mathcal{L}_{seg}$ is for text segmentation task in FCN, $\mathcal{L}_{ins}$, $\mathcal{L}_{reg}$ and $\mathcal{L}_{stop}$ represent losses for instance segmentation, bounding box regression and stop condition task in RNN respectively. $\lambda_{ins}, \lambda_{reg}$ and λ_{stop} are the hyper-parameters to control the balance among each task.

Loss for Text Segmentation. Text segmentation task in FCN uses the cross-entropy loss as the task can be regarded as a pixel-wise classification task. Denote the predicted value for a given pixel as y_i which is a 1D tensor of size 2 and $y_i^* \in \{0, 1\}$ is the ground truth. $\mathcal{L}_{seg}$ is give by

$$\mathcal{L}_{seg} = \frac{1}{N}\sum_{i\in\mathcal{L}_{seg}} \frac{-log(exp(y_i[y_i^*]))}{\sum_j exp(y_i[j])}. \quad (2)$$

In particular, some pixels don't produce any loss or gradient in two cases. First, text is taken as a positive sample only when its short side length ranges in $\left[32\times 2^{-1}, 32\times 2^{1}\right]$. If the short side length falls in $\left[32\times 2^{-1.5}, 32\times 2^{-1}\right) \cup \left(32\times 2^{1}, 32\times 2^{1.5}\right]$, we ignore this text instance, otherwise negative. Second, we enclose positive region with a "NOT CARE" boundary as transition from positive to negative. The boundary thickness is proportional to the short side length of text and the ratio is 0.1. In addition, we use the class balancing and hard negative sample mining introduced in [18] for better performance and faster loss convergence.

Loss for Text Instance Segmentation. Text instance segmentation task in RNN uses the IoU loss because the task is focused on distinguishing different instances and the areas of text regions vary tremendously. Denote the predicted map for a given sequence length n as $Y = \{Y_1, Y_2, ..., Y_n\}$ and $Y^* = \{Y_1^*, Y_2^*, ..., Y_{n^*}^*\}$ is the ground truth for a given image which has n^* instances.

Instead of imposing a specific instance order, we hope the model to find the optimal matching between the elements in Y and Y^* by itself. Therefore, we firstly construct a matching matrix M whose size is $\tilde{n} \times n^*$ and $\tilde{n} = min(n^*, n)$. The element in M is formulated as

$$M(t, t^*) = f_{IoU}(Y_t, Y_{t^*}^*), \quad (3)$$

$$f_{IoU}(Y_t, Y_{t^*}^*) = \frac{< Y_t, Y_{t^*}^* >}{||Y_t||_1 + ||Y_{t^*}^*||_1 - < Y_t, Y_{t^*}^* >}. \quad (4)$$

Eq.(4) is a relaxed version of the intersection over union (IoU) for the input which ranges in $[0, 1]$ in [23].

Then we put Y and Y^* in a bipartite graph and find the optimal matching matrix Δ by means of the Hungarian algorithm which is similar as [10]. Denote the $\Delta(i, j) = 1$ if and only if Y_i is assigned to Y_j^*, otherwise $\Delta(i, j) = 0$, $\mathcal{L}_{ins}$ is given by

$$\mathcal{L}_{ins} = -\sum_{t=1}^{\tilde{n}}\sum_{t^*=1}^{n^*} f_{IoU}(Y_t, Y_{t^*}^*)\Delta(t, t^*). \quad (5)$$

Loss for Text Bounding Box Regression. Text bounding box regression task in RNN uses the smooth L_1 loss because it is less sensitive to outliers than the L_2 loss according to [5]. Meanwhile, we only calculate the loss and gradient in the text region which is denoted as T. Denote the predicted value for a given pixel at the step t as z_i^t and $z_i^{t^*}$ is the ground truth which is assigned to z_i^t according to the optimal matching matrix Δ, $\mathcal{L}_{reg}$ is given by

$$\mathcal{L}_{reg} = \sum_{t=1}^{\tilde{n}}\sum_{i\in T}\left[z_i^{t^*} > 0\right] \cdot \text{smooth}_{L_1}(z_i^{t^*} - z_i^t), \quad (6)$$

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if} |x| < 1, \\ |x| - 0.5 & \text{otherwise.} \end{cases} \quad (7)$$

Fig. 5. Ground truth of each text instance. Top-left: original image. Top-right: ground truth for text segmentation. Bottom-left: ground truth for text instance segmentation. Bottom-right: ground truth for text bounding box regression.

Loss for Stop Condition. Stop condition task in RNN uses binary cross entropy loss because the task can be regarded as a binary classification task. If the number of iterations so far is equal or less than the total number of instances, we let the ground truth $s_t^* = 1$ at step t, otherwise $s_t^* = 0$. Denote the predicted probability is s_t at step t, $\mathcal{L}_{stop}$ is given by

$$\mathcal{L}_{stop} = \sum_{t=1}^{\tilde{n}} f_{BCE}(s_t^*, s_t), \tag{8}$$

$$f_{BCE}(a, b) = -(alog(b) + (1-a)log(1-b)). \tag{9}$$

Ground truth of each text instance is diagrammed in Figure 5. For each instance, we directly generate instance segmentation ground truth from quadrilateral coordinates and calculate the offset from coordinate of a quadrilateral vertex to each point in the text region as the bounding box regression ground truth which is similar as [1].

C. Extraction of Bounding Boxes

Traditional text detection algorithms need NMS and other post-processing. NMS is quite tedious when the number of bounding boxes is large or it needs to calculate an IoU among arbitrary quadrilaterals. As our method is from an instance segmentation perspective, we can extract each instance's bounding box in a much simpler way in the test phase. Denote the bounding boxes' scores at step t as S_t. The bounding box of the text instance is then obtained as the one with maximum score as

$$B_t = \underset{i}{\operatorname{argmax}}\, S_t(i). \tag{10}$$

IV. Experiments

A. Datasets

To compare our method with existing ones, we conduct experiments on two public benchmarks: ICDAR 2015 Incidental Scene Text Dataset [24] and ICDAR 2013 Focused Scene Text Dataset [25].

ICDAR 2015 Incidental Scene Text Dataset. There are 1000 training images and 500 test images in this dataset. It focuses on incidental scene text where the texts have various scales, resolution, blurring, orientations and viewpoints. In addition, each text word region is annotated by a quadrilateral with four corners.

ICDAR 2013 Focused Scene Text Dataset. There are 299 training images and 233 test images in this dataset, which focuses on the text content of interest. Meanwhile, all the text regions are annotated by horizontal rectangles.

TABLE I
Results on ICDAR 2015 Challenge 4 Incidental Scene Text Localization task.

Method	Recall	Precision	F-measure
Proposed	0.7554	**0.8560**	0.8026
He *et al.* [1]	**0.7968**	0.8234	**0.8099**
EAST [2]	0.7833	0.8327	0.8072
Yao *et al.* [29]	0.5869	0.7226	0.6477
Tian *et al.* [30]	0.5156	0.7422	0.6085
Zhang *et al.* [16]	0.4309	0.7081	0.5358

B. Implementation Details

To prepare the training samples, we use the datasets from ICDAR 2013 and ICDAR 2015. The image with size of 320×320 is cropped from images after random distortions like scaling and rotation. In the optimization stage, we first set the loss weight λ_{seg} to be 1 and other loss weights to be 0, then all loss weights are set to 1 after the segmentation task is well trained. For the training of RNN module, we adopt the curriculum learning by gradually increasing the number of objects that are required to be segmented and detected from the images. At the beginning, we only expect the network to extract at most 2 text words per image. After the training procedure converges, we increase this number to fine-tune the network until convergence, and keep iterating the process. At the test stage, the threshold of the stop condition is 0.5.

The whole network is optimized by Adam [26] and the initial learning rate is 0.001. When the training error plateaus, we multiply it by 0.1. All layers in the FCN model are initialized by xavier [27] and the rest layers in the recurrent structure are initialized at random, sampling them uniformly from the interval $[-0.08, 0.08]$. We conduct our experiments on 4 GPUS, with each GPU hosting 2 images (so the effective mini-batch size per iteration is 8 images). The whole experiments are implemented on the Lua/Torch deep learning frameworks [28] and run on a workstation with 2.9GHz 12-core CPU, 256G RAM, GTX Titan X and Ubuntu 64-bit OS.

C. Experimental Results

In Figure 6, we show some detection and instance segmentation results of our model. In these images we see that our model can handle various forms and numbers of text instances. The main errors are due to the difficulty to segment vague and perspective text lines.

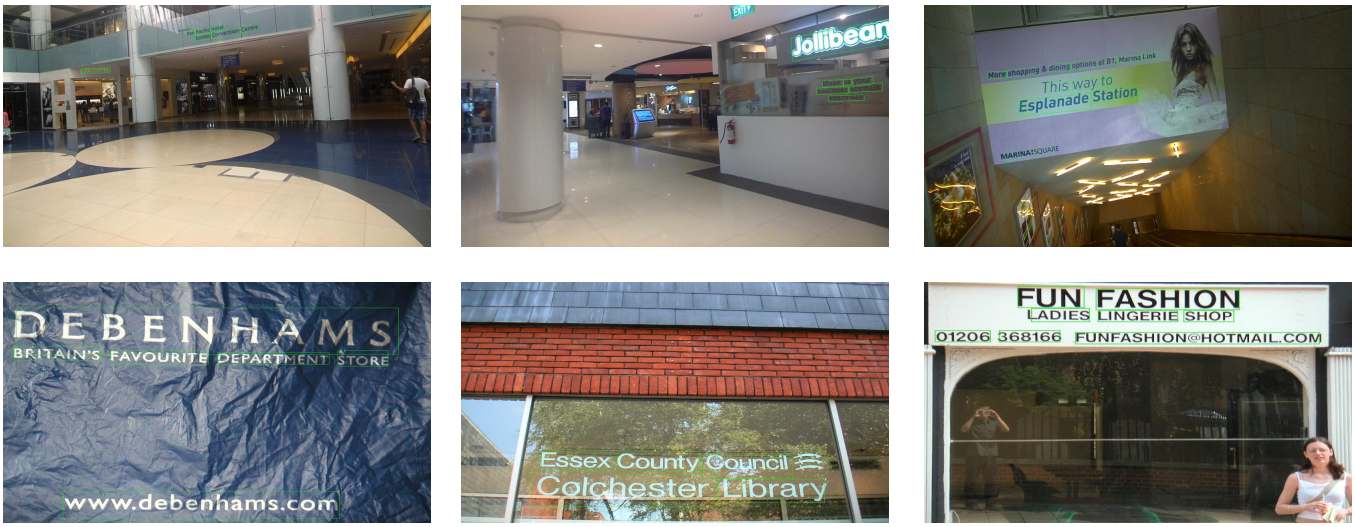

Fig. 6. Examples of detection results. First line: ICDAR 2015; second line: ICDAR 2013.

TABLE II
RESULTS ON ICDAR 2013 CHALLENGE 2 FOCUSED SCENE TEXT LOCALIZATION TASK.

Method	Recall	Precision	F-measure
Proposed	0.8595	**0.9513**	0.9031
RTN [31]	**0.8902**	0.9420	**0.9154**
Hu *et al.* [32]	0.8753	0.9334	0.9034
Tian *et al.* [30]	0.8298	0.9298	0.8769
Zhu *et al.* [33]	0.8164	0.9340	0.8713
He *et al.* [1]	0.81	0.92	0.86

As shown in Table I and Table II, the proposed method can achieve competitive results on ICDAR 2015 and ICDAR 2013. Meanwhile, our method can extract the bounding box in a simple way without NMS which greatly simplifies post-processing. Furthermore, our method outperforms previous methods in precision on ICDAR 2015 and ICDAR 2013 due to the stop condition branch can reduce some noise.

D. The Benefits of Regression Task

In order to further explore the benefits of regression task, we also evaluate a variant of our method when removing the bounding box regression branch. Although removing this branch, our method can still extract the bounding box of horizontal text line easily according to the instance segmentation results.

In Table III, model A is the baseline model which adopts the bounding box regression branch outputs. Model B extracts bounding boxes based on the results of model A's instance segmentation. Model C is the model which removes bounding box regression branch and extracts bounding boxes in the same way as model B.

TABLE III
ABLATION STUDIES ON THE ICDAR 2013 TEST SET.

Method	Recall	Precision	F-measure
model A	**0.8595**	**0.9513**	**0.9031**
model B	0.8539	0.9428	0.8962
model C	0.8283	0.9103	0.8674

The results of models B vs C shows that removing the bounding box regression branch reduces F1-measure score by 2.8%. This result indicates the benefits of the regression task to the instance segmentation task. The regression task constructs a multi-valued map rather than directly infers a binary mask, which encodes the boundaries of the text instance, so it can be regarded as a kind of boundary-aware segment prediction. Therefore, the regression task can make the instance segmentation results more accurate.

The results of models A vs B shows that obtaining the bounding box results directly from the instance segmentation results just reduces F1-measure score slightly by 0.7%. This indicates that the main role of regression task is describing the shape of each instance segmentation result. The improvement comes mainly from a few inclined text instances in ICDAR 2013.

V. CONCLUSION

In this paper, we propose a novel text detection method based on FCN and RNN to perform both detection and instance segmentation. The FCN is firstly used to classify text and non-text regions, and then the RNN utilizes the features

extracted by FCN to simultaneously detect and segment one text instance at each time step. Meanwhile, it can extract bounding boxes directly which is conceptually simpler than existing methods relying on NMS. Our experiments on ICDAR 2015 and ICDAR 2013 have demonstrated that our method is competitive to the state-of-the-art methods and performs better in accurate location of text instances. We also evaluate a variant of our method in order to explore the benefits of regression task. We believe this work could provide more insight and promotion to scene text detection.

ACKNOWLEDGEMENT

This work has been primarily supported by National Natural Science Foundation of China (NSFC) Grants 61721004, 61411136002 and 61733007.

REFERENCES

[1] W. He, X. Y. Zhang, F. Yin, and C. L. Liu, "Deep direct regression for multi-oriented scene text detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 745–753.

[2] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: An efficient and accurate scene text detector," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2642–2651.

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

[4] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," in *Proceedings of the Association for the Advance of Artificial Intelligence*, 2017, pp. 4161–4167.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks." in *Proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, 2017, pp. 1137–1149.

[6] X. Liang, Y. Wei, X. Shen, Z. Jie, J. Feng, L. Lin, and S. Yan, "Reversible recursive instance-level object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 633–641.

[7] S. Liu, X. Qi, J. Shi, H. Zhang, and J. Jia, "Multi-scale patch aggregation (mpa) for simultaneous detection and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3141–3149.

[8] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.

[9] K. He, G. Gkioxari, P. Dollr, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[10] R. Stewart, M. Andriluka, and A. Y. Ng, "End-to-end people detection in crowded scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2325–2333.

[11] B. Romera-Paredes and P. H. S. Torr, "Recurrent instance segmentation," in *Proceedings of the IEEE Conference on European Conference on Computer Vision*, 2016, pp. 312–329.

[12] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[13] S. Tian, Y. Pan, C. Huang, and S. Lu, "Text flow: A unified text detection system in natural scene images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4651–4659.

[14] T. He, W. Huang, Y. Qiao, and J. Yao, "Text-attentional convolutional neural network for scene text detection," in *Proceedings of the IEEE transactions on image processing*, vol. 25, no. 6, 2016, pp. 2529–2541.

[15] Z. Zhang, W. Shen, C. Yao, and X. Bai, "Symmetry-based text line detection in natural scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2558–2567.

[16] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4159–4167.

[17] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Proceedings of the IEEE Conference on European conference on computer vision*, 2016, pp. 21–37.

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

[19] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4438–4446.

[20] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan, "Proposal-free network for instance-level object segmentation," in *Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, pp. 1–1.

[21] Z. Zhang, A. G. Schwing, S. Fidler, and R. Urtasun, "Monocular object instance segmentation and depth ordering with cnns," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2614–2622.

[22] A. Graves, "Long short-term memory," in *Proceedings of the Neural computation*, vol. 9, no. 8, 1997, pp. 1735–1780.

[23] V. Koltun, "Parameter learning and convergent inference for dense random fields," in *Proceedings of the International Conference on Machine Learning*, 2013, pp. 513–521.

[24] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, and S. Lu, "Icdar 2015 competition on robust reading," in *Proceedings of the International Conference on Document Analysis and Recognition*, 2015, pp. 1156–1160.

[25] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. I. Bigorda, S. R. Mestre, J. Mas, and D. F. Mota, "Icdar 2013 robust reading competition," in *Proceedings of the International Conference on Document Analysis and Recognition*, 2013, pp. 1484–1493.

[26] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *arXiv preprint arXiv:1412.6980*, 2014.

[27] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[28] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," in *Proceedings of the Advances in Neural Information Processing Systems Workshop*, 2011.

[29] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao, "Scene text detection via holistic, multi-channel prediction," in *arXiv preprint arXiv:1606.09002*, 2016.

[30] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proceedings of the IEEE Conference on European Conference on Computer Vision*, 2016, pp. 56–72.

[31] X. Zhu, Y. Jiang, S. Yang, X. Wang, W. Li, P. Fu, H. Wang, and Z. Luo, "Deep residual text detection network for scene text," in *Proceedings of the International Conference on Document Analysis and Recognition*, 2017, pp. 807–812.

[32] H. Hu, C. Zhang, Y. Luo, Y. Wang, J. Han, and E. Ding, "Wordsup: Exploiting word annotations for character based text detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4950–4959.

[33] S. Zhu and R. Zanibbi, "A text detection system for natural scenes with convolutional feature learning and cascaded classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 625–632.