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

Scene Text Detection via Deep Semantic Feature Fusion and Attention-based Refinement

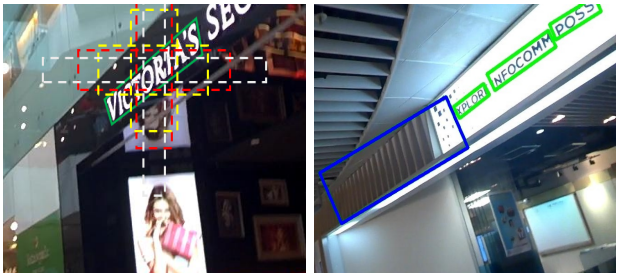
Yu Song[+,1,2], Yuanshun Cui[+,1,2], Hu Han[*,1], Shiguang Shan[1,2] and Xilin Chen[1,2]
[1]Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
{yu.song, yuanshun.cui}@vipl.ict.ac.cn, {hanhu, sgshan, xlchen}@ict.ac.cn

***Abstract*—Despite tremendous progress in scene text detection in the past few years, efficient text detection in the wild remains challenging, particularly for the texts have large rotations, and the complicated background areas that are easily confused with text. In this paper, we propose an effective approach for scene text detection, which consists of initial text detection using the proposed deep semantic feature fusion of a fully convolutional network (FCN), and text detection refinement by our attention based text vs. non-text classifier learned in a fine-to-coarse fashion. The proposed approach outperforms the state-of-the-art scene text detection algorithms on the public-domain ICDAR2015 dataset, achieving an accuracy of 0.83 in terms of F-measure.**

I. Introduction

In the past few years, scene text detection and recognition have attracted increasing attentions because of their great potentials for practical applications and the challenges of establishing the fundamental methods. Nevertheless, owing to the limitations of conventional image processing methods, text detection which mainly use traditional approaches to segment text regions from the background cannot achieve significant improvement. Recently, the performance of scene text detection has been significantly improved benefited from the general-purpose object detection methods that use deep learning, such as Faster-RCNN [1], SSD [2] and R-FCN [3].

However, even these deep learning based text detection methods have their limitations. For example, as shown in Fig. 1(a), while the anchor mechanism in Faster R-CNN is effective for generating proposals using region proposal network (RPN) for objects that are usually upright in the images, such proposals become less effective in covering the texts that are usually neither horizontal nor vertical due to arbitrary orientations during image acquisition [4]. Even though some methods have tried to improve the proposal-generating mechanisms to handle the multi-oriented texts [5], there are still inherent drawbacks, such as inefficiency of RPN. By contrast, the recent R-FCN based detection method does not rely on anchor-based proposal generation, and therefore could better handle the irregular text orientations. However, it is still very difficult for R-FCN to robust detect the texts in the wild that have large-scale variations. Besides, false positive text region as shown in Fig. 1(b) may appear when the complicated background can be easily confused with the text. We argue that the false positive is affected by the deceptive appearance of that region. The study in [6] shows the possibility of refining detection through recognition, but its performance still suffers from the anchor mechanism of the backbone network, and the manner of using refinement.



(a) (b)

Fig. 1: Text detection in the wild is challenging because of (a) the large rotations of the texts, and (b) the complicated background that can be easily confused with text at distance (e.g., the background in the blue bounding-box can be easily confused with character 'm'.)

In this paper, to address the above issues, we present a novel framework for detecting arbitrarily oriented scene texts. The proposed approach consists of an initial text detection using the proposed deep semantic feature fusion of a fully convolutional network, and text detection refinement by our attention based text vs. non-text classifier learned in a fine-to-coarse fashion. Our approach is especially useful for detecting scene text which is particularly difficult due to the complex background, arbitrary orientation, various scales and extreme light conditions, etc. The contributions of this paper are three-fold: (i) semantic feature fusion based on an FCN [7] network, leading to robustness against text variations in scale, illumination, and orientation; (ii) attention-based text vs. non-text refinement learned in a fine-to-coarse fashion leading to efficient removal of complicated false detections without sacrificing the detection precision, and (iii) state-of-the-art accuracy on the challenging ICDAR2015 dataset.

II. Related Work

Scene text reading is a particularly important field of computer vision, which aims to extract textual information

[+]equal contribution;[*]H. Han is the corresponding author.

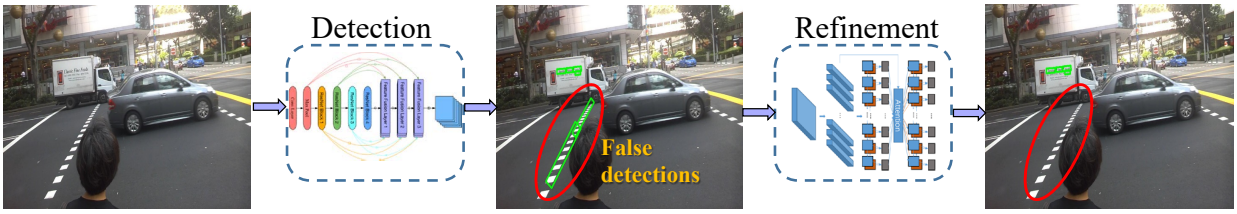


Fig. 2: The pipeline of the proposed scene text detection method, consisting of an initial text detection followed by an attention-based text vs. non-text refinement.

from images or videos. In the existing methods, scene text reading usually contains two tasks: text detection and text recognition. While text detection addresses the problem of localizing the regions containing text from an input image, text recognition is to "translate" a text region into characters and words. Apparently, scene text detection is the preceding task for the scene text recognition. We focus on text detection in this paper, but still provide a brief review for methods of both scene text detection and recognition.

Scene text detection. The early methods on text detection treated the text region as connected components [8], [9], and tried to find connected regions with the characteristics of text. For example, Stoke Width Transform (SWT) [8] extracted the text region by detecting the text edges, and a Maximally Stable Extremal Regions (MSER) based method like [10] was used to find the extremal regions, which were considered as the text regions. Recently, deep learning based methods have been widely used for scene text detection methods [4], [5], [6], [11]. For example, the method in [5] proposed to generate rotated proposals by considering the arbitrary text rotations. The method in [6] used a modified SSD network [2] to better handle the aspect variations appeared in scene text. They also proposed a refinement mechanism that utilizes the recognition score to improve the text detection accuracy. The method in [11] and [4] used a FCN-based network and modified the traditional non-maximum suppression method to detect scene text.

Scene text recognition. There are a number of scene text recognition methods in the literature [12], [13], [14], [15]. [12] presented a comprehensive survey of text detection, tracking, and recognition in video. The method in [15] detected individual characters using sliding window, and then use lexicon search to decode the text. The method in [13] proposed to recognize the characters in each sliding window first, and then decode these characters using a language model. The method in [14] combined Convolution Neural Network (CNN) with Long Short-Term Memory networks (LSTM), and treated images as a sequence to "translate" it to text. Their method reported the state-of-the-art performance on the four popular databases [15], [16], [17], [18]. In addition, there are also a number of studies for Chinese text recognition [19], [20].

Similar to [6], our approach takes the merits of the framework of text detection followed by detection refinement However, our approach differs from existing methods in that: (i) our method uses deep semantic fusion to achieve robustness against the large diversity in text scale; (ii) we propose an attention-based refinement to perform accurate text vs. non-text accuracy classification with a fine-to-coarse training.

III. Proposed Approach

A. Formulation

The objective of our method is to detect the text region from an input image captured under a complex scene, and the formulation is given by

$$y = \mathcal{R}(\mathcal{D}(I)), \tag{1}$$

where I is an input image and $y = (L, P)$ is the output text location. $\mathcal{D}$ is our initial text detection function defined as

$$[U, V] = \mathcal{D}(I), \tag{2}$$

where U and V are the detected text location by $\mathcal{D}$ and the corresponding detection confidence. $\mathcal{R}$ is our text refinement function, which takes the detection outputs by $\mathcal{D}$ and gives a text vs. non-text classification score, i.e., $y = \mathcal{R}([U, V])$

In this work, we use a revised FCN network to extract features and then fuse semantic features to perform the text detection corresponding to $\mathcal{D}$, and design an attention-based text vs. non-text classifier to eliminate the false detections. The diagram of the proposed scene text detection method is illustrated in Fig. 2.

B. Scene Text Detection

FCN was originally proposed for the segmentation tasks, but has been found to have big potential in object detection tasks. In [4] and [11] FCN is efficient in text detection. However, FCN based text detection still have some limitations in detect text under complicated scenarios (see Fig. 1). To handle the complicated background and extremely varied text scales, we argue that the semantic features of FCN with different scales should be utilized. Inspired by [21], we proposed a deep semantic feature fusion method to do scene text detection. Our detection method includes three main parts: semantic feature fusion, multi-channel regression, and Non-Maximum Suppression (NMS).

Detection Network Architecture. To get the text region, the features of an input image are extracted by using a ResNet [22], and then the multi-scale semantic features are fused to improve the perception ability of the model for text regions with diverse scales. Then, the offsets of each pixel are estimated and matched with the text region through the mask score. Finally, after obtaining the text bounding-box, NMS is applied to filter the redundant and outlier detections.

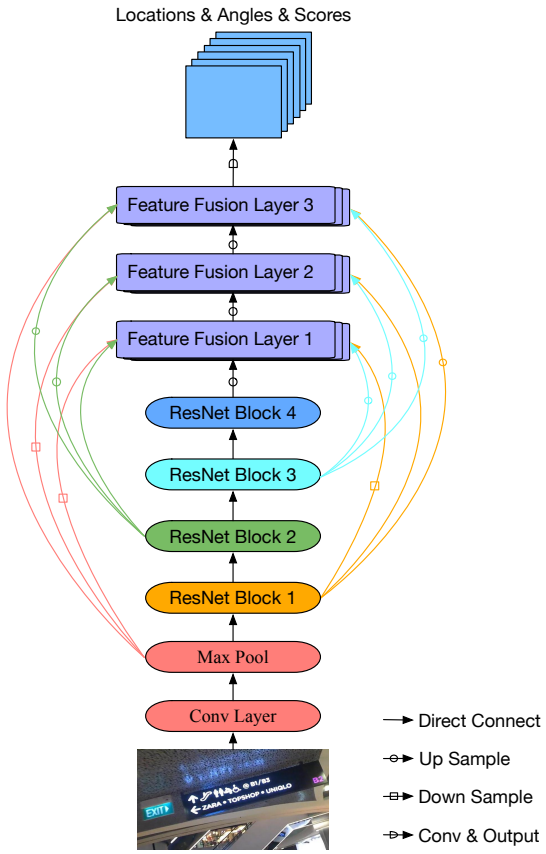


Fig. 3: The architecture of our detection model using deep semantic feature fusion.

We use a ResNet-50 [22] as the base feature extraction module because of its high efficiency and representation ability. The detailed architecture of our model is shown in Fig. 3. The four ResNet blocks contain 3, 4, 6, 3 "bottleneck" building blocks, respectively. We use the output features of the four ResNet segments for multi-scale semantic feature fusion. Inspired by [21], we adopt the densely deep semantic feature fusion mechanism so that the features for detection are able to capture the text regions with different scales and shapes. Notably, we would upsample and downsample the features from the preceding stages to match the feature map sizes in the latter stages.

Loss function. To take the text confidence score and text localization error into consideration together, we follow [4], [11] and use a modified dice loss function [23].

$$\begin{cases} L &= L_s + L_{loc} \\ L_s &= 1 - \dfrac{2 \cdot \mathbf{S_P} \odot \mathbf{S_G}}{\mathbf{S_P} \odot \mathbf{S_G} + ||\mathbf{S_G}||_2^2} \\ L_{loc} &= -\log \dfrac{R_p \cap R_g}{R_p \cup R_g} + 1 - \cos(\theta_p - \theta_g) \end{cases} \tag{3}$$

where $\mathbf{S_P}$ and $\mathbf{S_G}$ are the estimated score map and ground-truth score map while $\odot$ is an operation that sum the result of Hadamard product. R_p and θ_p represent the offsets and rotation angle of an estimated text region, while R_g and θ_g represent the ground-truth region location and its rotation angle. By using the modified dice loss function, our detection model could handle the class imbalance between text and non-text regions. In addition, it converge faster.

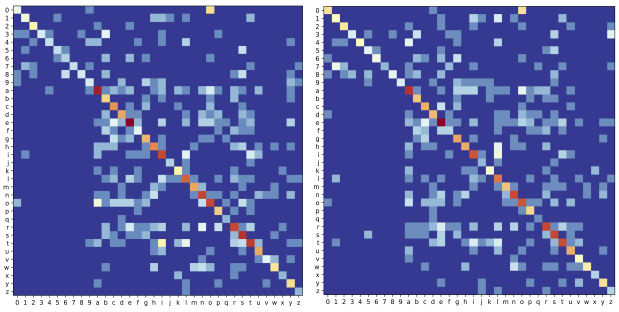
Non-Maximum Suppression(NMS). The text regions predicted by the detection network are usually intensively overlapped due to the similar scores between the adjacent text regions. Therefore, a post-process step using NMS is applied similar to the other object detection tasks. To get better post-processing results without sacrificing the processing speed, we experimented with several recent NMS methods such as Recalled NMS [4] and Locality-Aware NMS [11], etc. We found that these modified NMS methods usually contain several additional components than the traditional NMS, such as merging results (combine adjacent text regions by weighting their scores) and enveloping results (find the minimum enclosing bounding box of several different regions). We finally choose to use NMS with additional merging.

C. Detection Refinement via Attention-based Classification

As the detection network will inevitably generate false positive text detections, given the strong correlation between text detection and recognition tasks, we argue that a recognition network could be helpful for refining the detection results. We propose a novel attention-based text vs. non-text refinement model based on a revised CRNN (named CRANN). **CRANN.** CRNN [14] is an effective method for text recognition from a text detection image region. However, we find that some characters could be easily confused with each other in CRNN. The confusion matrix of CRNN could explicitly address the problem. For example, Fig. 4(a) shows a confusion matrix of the characters when using the original CRNN. The brightness of each element denotes the probability of grouping one character into another. For example, the element corresponding to 'i' and 't' is particularly high, which means these two characters are easy to be confused. The situations are also observed for '0' and 'o,' 'a' and 'o,' etc. Although our text vs. non-text refinement does not require decoding a detected text region into characters. The features that are informative for differentiating individual characters are also helpful for distinguishing text and non-text. Generally speaking we hope to learn strong features that are informative for differentiating individual characters, and then transfer these features for the binary text vs. non-text classification.

Let the encoder layer and decoder layer denote the two Bi-LSTM layers in CRNN, respectively. We argue that the main reason for the above confusions is caused by the innate mechanism of the encoder layer. Specifically, while the features at the early time steps may contain information to avoid the confusion. it is ignored due to its weight decay.

Thus, to extract those information, we propose the CRANN by revising the attention mechanism [24] and integrating it into



(a) CRNN (b) CRANN

Fig. 4: Confusion matrices of the original CRNN [14] and the proposed CRANN for detection refinement, which are visualized on IIIT5K [16] at the fine stage during the fine-to-coarse training. The wrongly decoded characters are counted to draw the confusion matrices.

CRNN. The revised attention mechanism can be formulated as

$$\begin{cases} \mathbf{g}_{ij} = \tanh\left(\mathbf{H}_i + \mathcal{L}(\mathbf{F}_j)\right), \quad i,j = 1,2,\ldots,T \\ e_{ij} = \mathcal{L}_e(\mathbf{g}_{ij}) \\ \alpha_{ij} = \dfrac{\exp(e_{ij})}{\sum_{k=1}^{T}\exp(e_{ik})} \\ \mathbf{c}_i = \sum_{j=1}^{T}\alpha_{ij}\mathbf{F}_j \end{cases} \tag{4}$$

where T is the sequence length. $\mathbf{H}$ and $\mathbf{F}$ are the encoder Bi-LSTM output and the feature sequence, respectively. $\mathcal{L}$ is a linear transformation that projects $\mathbf{F}$ to the space of $\mathbf{H}$. $\mathcal{L}_e$ is also a linear transformation for projecting the vector $\mathbf{g}_{ij}$ to a real number. $\mathbf{c}_i$ is a weighted sum of the image features.

The detailed architecture of our revised CRANN (named CRANN) is shown in Fig. 5. The proposed CRANN outperforms the CRNN on several popular benchmarks under the same testing protocols [14]. For example, our CRANN model outperforms CRNN on IIIT5K [16] without lexicon (81.2% vs. 80.9%). In addition, our CRANN does reduce the confusion between a number of characters (see Fig. 4(b)).

Finally, we modify the transcription and output layers of CRANN into a text vs. non-text classification layer. The final refinement module determines if an input image has text or not, and gives its confidence.

Fine-to-Coarse Training. As discussed above, the informative features for differentiating individual characters can be helpful for our text vs. non-text classification task. Therefore, we adopt a fine-to-coarse training strategy to learn our refinement network. First, we train our CRANN model on a public-domain dataset. Second, we train the CRANN model for a fine-scale character recognition task on a generated dataset which contains a wide range of characters to expand its decoding ability. Then, with the weights from the pre-trained of CRANN, we fine-tune it for a two-class text vs. no-text classification using a large number of image regions containing both text and non-text.

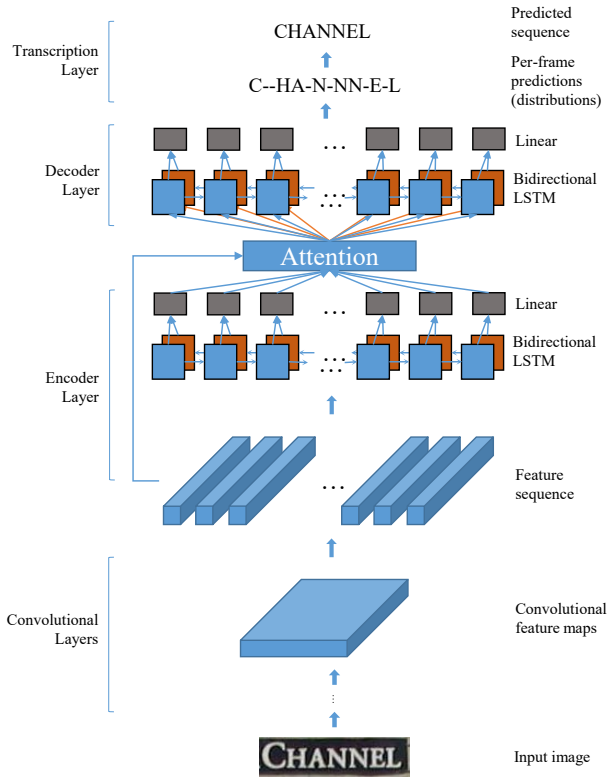


Fig. 5: The detailed architecture of our CRANN model for text vs. non-text classification.

Refinement Methodology. As shown in Fig. 2, we first perform text detection using the initial text detection module. The initial text bounding-boxes are then sent into the refinement module for text vs. non-text classification. As described in Eqs.(1) and (2), the refinement module takes a cropped text image as input and generates a text vs. non-text confidence score. We empirically choose to use 0.998 as the threshold to determine an input as text.

IV. Experimental Results

We evaluate the proposed approach on the public-domain ICDAR2015 database [25], and provide comparisons with a number of the state-of-the-art scene text detection algorithms.

A. Databases and Experimental Settings

SynthText dataset [26] dataset consists of 800K images with approximately 8 million synthetic word instances that has abundant characters. Synthetic [27] consists of 9 million images which cover 90K English words. IIIT5K dataset [16] dataset contains 3,000 cropped test images of word collected from the Internet. SVT dataset [15] dataset contains 249 images collected from Google Street View. ICDAR2003 [17] test dataset contains 251 scene images. ICDAR2013 [18] contains 229 training images and 223 test images, which were mainly horizontal scene texts. ICDAR2015 [25] is a benchmark of the ICDAR2015 Robust Reading Competition and is the most widely used dataset in recent publications, which contains 1,000 training images and 500 test images. Due to the small database size in the field of text detection, in the

following, we combine several public datasets for pre-training.

Training Data Generation. To train the detection network, we follow the data pre-processing methodology introduced in [11] to prepare our training dataset. First, we mask off non-text region and compute the offsets of each pixel that lies in the text region as well as the rotation angle of each text region. Therefore, each predicted text region could be formulated as

$$\begin{cases} \mathbf{T_{pred}} = \mathbf{M_G} * \mathbf{M}_\Delta \\ M_G^{(i,j)} = \begin{cases} 0, \text{ for } (i,j) \in \text{ Text Region} \\ 1, \text{ for } (i,j) \notin \text{ Text Region} \end{cases} \\ M_\Delta^{(i,j)} = \{(\Delta t_{ij}, \Delta r_{ij}, \Delta d_{ij}, \Delta l_{ij}, \Delta\theta_{ij}) | (i,j) \in \mathbf{I}\} \end{cases} \quad (5)$$

where $(\Delta t_{ij}, \Delta r_{ij}, \Delta d_{ij}, \Delta l_{ij}, \Delta\theta_{ij})$ represents the offsets of each pixel in image $\mathbf{I}$ that lies in the text region and indicated by the $\mathbf{M_G}$ w.r.t. the text region boundaries (we assume that a text region could be enveloped by four straight lines).

Due to the limited size of the ICDAR2015 dataset, deep learning based method could easily get overfitting. Similar to the published methods [11], [14], the public-domain SynthText and Synthetic datasets are used to pre-train our text detection model and refinement model. Specifically, we use the whole SynthText dataset to pre-train our detection network for 100K iterations with a batch size of 16. We then fine-tune the the detection network using a combination of training set of the ICDAR2015 and the training set of ICDAR2013 for another 10K iterations with a batch size of 16. Finally, we evaluate our detection method on the ICDAR2015 test dataset.

For the refinement network, we first trained the proposed CRANN on Synthetic following the method in CRNN [14]. Then, we train CRANN on the randomly cropped images from SynthText (called SSub, approximately 60% of word instances are from SynthText), and change the alphabet from alphanumeric characters alone to alphanumeric and punctuation characters to recognize a wider range of characters. Finally, we modify the output layer of the pre-trained CRANN into two classes, and train our text vs. non-text classification model on SSub and randomly cropped background non-text images from the ICDAR2015 training dataset.

B. Effectiveness of Individual Modules

To verify the effectiveness of our individual components in our approach, we first evaluate our detection network alone on the ICDAR2015 test set. Our detection network alone could obtain 0.82 of F-measure, which outperforms the state-of-the-art method in [4].

The effectiveness of our CRANN refinement model is validated on four public-domain text recognition datasets(IIIT5K, SVT, ICDAR2003, ICDAR2013) following the testing protocol in [14]. The proposed approach outperforms CRNN on all the databases. For example, on IIIT5K, when decoding without the lexicon, our CRANN could reach 81.2% accuracy, while the accuracy of the original CRNN in [14] is 80.9%. Furthermore, the final detection refinement model could achieve 91.8% accuracy for text vs. non-text classification on a combination dataset of ICDAR2015 recognition dataset and Internet background images.

Fig. 6: Examples of the text detection results by the proposed approach without (left) and with (right) using the attention-based detection refinement.

Finally, when tested with integrated the detection network and refinement model, our approach obtains 0.83 of F-measure which outperforms the state-of-the-art. Fig. 6 gives some examples of how the proposed detection refinement helps to improve the text detection accuracy. As shown in Fig. 6, the left column shows the text detection results by our detection network alone. It shows that our detection network can capture diverse complex scene texts, such as texts with bad illumination, low-resolution, arbitrary inclines, and diverse scales. The right column shows the text detection results after using the proposed refinement. Compared with the initial text detection results, our refinement model demonstrates its ability in eliminating false positive detections that are easily confused with text regions. Therefore, the precision of detection results are further improved.

C. Comparisons with the State-of-the-art

We compare the proposed text detection method with a number of the state-of-the-art text detection methods in the literature, such as [4], [5], [9], [11], [25], [28], on the ICDAR2015 dataset. The detection results by individual methods are reported in Table I. As shown in Table I, our text detection method outperforms the best of the state-of-the-art methods (i.e. [4]) in terms of F-measure (0.83 vs. 0.81) on ICDAR2015. We also evaluated our method on ICDAR2013. It achieves 76.2% and 76.3% before and after refinemnet, which are comparable to some of the state-of-the-art methods [29], [30].

The methods in [4], [11] adopted FCN-based pipeline to detect text target directly. However, they still have limitations in taking full advantages of the semantic features in different stages and eliminating false positive detections that are easily

TABLE I: Comparisons of the proposed approach and a number of the state-of-the-art text detection methods on the ICDAR2015 dataset.

Algorithm	Precision	Recall	F-measure
HUST_MCLAB [25]	0.44	0.38	0.41
AJOU [25]	0.47	0.47	0.47
NJU-Text [25]	0.70	0.36	0.47
StradVision1 [25]	0.53	0.46	0.50
StradVision2 [25]	0.77	0.37	0.50
Zhang et al. [28]	0.71	0.43	0.54
Tian et al. [9]	0.74	0.52	0.61
Liu ea al. [5]	0.73	0.68	0.71
Zhou et al. [11]	0.78	**0.83**	0.80
He ea al. [4]	0.82	0.80	0.81
Proposed (Without Refinement)	0.84	0.80	0.82
Proposed (With Refinement)	**0.86**	0.80	**0.83**

confused with text regions. By contrast, the proposed deep semantic feature fusion with refinement used for text detection is able to overcome these limitations.

D. Computational Cost

We profile the proposed approach on the ICDAR2015 dataset on a desktop with NVIDIA Titan X GPU and Intel i7-3.60GHz CPU. For text detection part using deep semantic features, it takes about 70ms on average; for detection refinement part via attention-based classification, it takes about 0.08ms on average. So overall text detection by the proposed approach takes less than 80ms per image on average.

V. Conclusions

In this paper, we have proposed a new scene text detection approach consisting of text detection using deep semantic feature fusion and refinement with an attention-based model. By using a deep semantic feature fusion, our initial text detection network can achieve robustness against variations of bad illumination, diverse scales, and arbitrary rotations. The proposed attention based refinement network leverages a fine-to-coarse training scheme to learn informative features for differentiating text from non-text detections. Evaluations on the public-domain ICDAR2015 dataset show that the proposed text detection approach outperforms the state-of-the-art methods, and achieves an accuracy of 0.83 in terms of F-measure. In our future work, we would like to study the feature representations that are helpful for detecting curved text.

Acknowledgment

This research was partially supported by Natural Science Foundation of China (grants 61732004, 61672496, and 61650202), Strategic Priority Research Program of CAS (grant XDB02070004), and Youth Innovation Promotion Association CAS (grant 2018135).

References

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

[2] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *ECCV*, 2016.

[3] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," in *NIPS*, 2016.

[4] W. He, X.-Y. Zhang, F. Yin, and C.-L. Liu, "Deep direct regression for multi-oriented scene text detection," in *ICCV*, 2017.

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

[6] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network." in *AAAI*, 2017.

[7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[8] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *CVPR*, 2010.

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

[10] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *CVPR*, 2012.

[11] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: An efficient and accurate scene text detector," in *CVPR*, 2017.

[12] X. Yin, Z. Zuo, S. Tian, and C.-L. Liu, "Text detection, tracking and recognition in video: A comprehensive survey," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2752–2773, 2016.

[13] F. Yin, Y.-C. Wu, X.-Y. Zhang, and C.-L. Liu, "Scene text recognition with sliding convolutional character models," *arXiv:1709.01727*, 2017.

[14] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 11, pp. 2298–2304, 2017.

[15] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *ICCV*, 2011.

[16] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *BMVC*, 2012.

[17] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young, K. Ashida, H. Nagai, M. Okamoto, H. Yamamoto *et al.*, "Icdar 2003 robust reading competitions: entries, results, and future directions," *IJDAR*, vol. 7, no. 2, pp. 105–122, 2005.

[18] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i. Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazàn, and L. P. de las Heras, "ICDAR 2013 robust reading competition," in *ICDAR*, 2013.

[19] R. Dai, C.-L. Liu, and B. Xiao, "Chinese character recognition: history, status and prospects," *Frontiers of Computer Science in China*, vol. 1, no. 2, pp. 126–136, 2007.

[20] X.-Y. Zhang, Y. Bengio, and C.-L. Liu, "Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark," *Pattern Recognition*, vol. 61, pp. 348–360, 2017.

[21] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," in *CVPR*, 2017.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[23] F. Milletari, N. Navab, and S. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *3DV*, 2016.

[24] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[25] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. 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 *ICDAR*, 2015.

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

[27] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *IJCV*, vol. 116, no. 1, pp. 1–20, Jan. 2016.

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

[29] S. Lu, T. Chen, S. Tian, J.-H. Lim, and C.-L. Tan, "Scene text extraction based on edges and support vector regression," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 18, no. 2, pp. 125–135, 2015.

[30] X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao, "Robust text detection in natural scene images," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 5, pp. 970–983, 2014.