Scene Text Relocation with Guidance

Anna Zhu
School of Computer
Wuhan University of Technology
Wuhan, China
Email: annakkk@live.com

Seiichi Uchida
Human Interface Laboratory
Kyushu University
Fukuoka, Japan
Email: uchida@ait.kyushu-u.ac.jp

***Abstract*—Applying object proposal technique for scene text detection becomes popular for its significant improvement in speed and accuracy for object detection. However, some of the text regions after the proposal classification are overlapped and hard to remove or merge. In this paper, we present a scene text relocation system that refines the detection from text proposals to text. An object proposal-based deep neural network is employed to get the text proposals. To tackle the detection overlapping problem, a refinement deep neural network relocates the overlapped regions by estimating the text probability inside, and locating the accurate text regions by thresholding. Since the space between words in different text lines are various, a guidance mechanism is proposed in text relocation to guide where to extract the text regions in word level. This refinement procedure helps boost the precision after removing multiple overlapped text regions or joint cracked text regions. The experimental results on standard benchmark ICDAR 2013 demonstrate the effectiveness of the proposed approach.**

I. Introduction

Text, as an important and intuitive visual object in natural scene images, is beneficial for content-based image understanding and analysis. The relevant research [1] has been used on many applications, such as license plate recognition, automatic translation, robot navigation, virtual reality, etc. Text detection is the initial and promising step in general processing pipeline for its accuracy has extremely influence on the sequential text recognition.

Scene text detection is a high level visual task, which is difficult to be solved completely by a set of low-level operations or manually designed features. Up to very recently, the deep learning applied object classification and detection methods achieved remarkable performance. The deep neural network (DNN) is capable of learning meaningful high-level features and semantic representations for visual recognition through a hierarchical architecture with multiple layers of feature convolutions. The trend that sample, classify and regress the object proposals from a set of default boxes on every feature map location can effectively increase the detection accuracy and decrease the computational cost. In this paper, we use the object proposals techniques and the DNN to locate the text in natural scene images. It produces a set of good quality text bounding boxes with high recall by classifying sampled regions in a deep convolutional neural network (CNN). Similar to sliding window-based methods, it also results in overlapping problem of the detected text regions. The non-maximum suppression (NMS) process has difficulty to eliminate the situations as follows.

(a) (b) (c)

Fig. 1. Overlap situations of text detection results.

(a) A word region is partly located with one more overlapped bounding boxes of high and approximate text confidence (Fig. 1(a)).

(b) Neighbored words aligning closely are located as one word (Fig. 1(b)).

(c) Vertical neighbored words are partly detected as a text region (Fig. 1(c))

To tackle this problem, a deep neural network combined of CNN and Long Short Term Memory (LSTM) network is cascaded to refine the text detection results. Given an enlarged regions, namely the convex rectangle region containing two overlapped text proposals in the image, this refinement network aims to return the accurate location of text regions inside. This relocation process can efficiently remove the overlapped regions or connect cracked regions, thus improve the detection precision rate drastically. The major contributions of our work are claimed as follows.

Firstly, we developed a two-stage text detection system which can maintain the high recall of the initial text proposal stage while boosting precision in the refinement stage. No grouping step or post-processing are required in our text detection system. Therefore, it is more efficient and faster.

Secondly, a text relocation network with guidance was proposed to predict the accurate text regions of each row then sequentially each column in the overlapped rectangle regions. This relocation process can help to improve the text detection precision.

Finally, we evaluated both the text detection system with and without the refinement network on benchmark dataset ICDAR2013. The experimental results demonstrated our proposed method can improve the precision extremely. Compared

with current state-of-the-art text detection methods, it also showed its superiority.

The rest of the paper is structured as follows: A selection of related work is reviewed in Sect. II. Sect. III presents our proposed method in detail. In Sect. IV, we give the experimental results which include the details of databases and the experimental setup. Finally, Sect. V gives a summarization and conclusion of this paper.

II. Related work

In this section, some previous scene text detection work are reviewed. Roughly, we group them to two categories: connected component analysis (CCA)-based and regional-based approaches.

The CCA-based methods detect text from pixel-level to character level, then to text line level sequentially. Traditional CCA-based approaches [2] detect text by low level features and heuristic rules. Since CNN has the good ability to represent text features, it is widely incorporated with the CCA-based methods for text/non-text classification [3], [4] on candidate character components. Those approaches extract character or text components by exploring low-level image cues which is not robust. Incidently, a large amount of non-text components are generated. Even counting on CNN-based classifiers, it is not easy to filter out them.

Building on recent advances of deep learning models for image representation, many researches used CNN for pixel-wised text regions extraction directly. It casts text detection as segmentation problem [5] to predict the text probability of each pixel by Fully Convolutional Network(FCN)-based models [6]. And then perform MSER or graph cut on salient text regions to get text components. High-level features are extracted directly from the whole images for text regions detection. Algorithms directly runs on full images and few post-processing steps are required. Compared with the traditional CCA-based method, they are more robust and faster.

Regional-based methods usually adopt sliding window strategy, casting the text detection as a classical object detection task. By scanning the images with multi-scale windows, discrete sub-image space regions are captured, and then classified through texture classifiers or CNN [7], [8]. Extracting features for each region independently is identified as the bottleneck in these exhaustive region searching manner.

The object proposals techniques which share convolutions across proposals [9], [10] and carried on with the DNNs emerge as an alternative to the traditional regional-based text detection approaches. Zhong et al. [11] designed a inception region proposal network to achieve only hundred level candidate text proposals from a set of text characteristic prior bounding boxes. Those candidate text regions were further classified and regressed for accurate localization by a text detection network that embedded ambiguous text category information and multilevel region-of-interest pooling. Gupta et al. [12] proposed a fully-convolutional regression network, which draw on the image-grid based bounding box regression network YOLO [13], to perform text detection at all locations

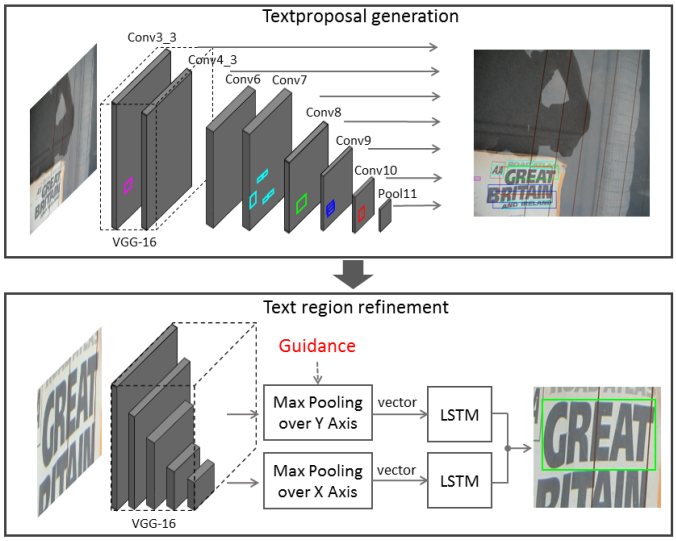


Fig. 2. The pipeline of the proposed text detection system.

and multiple scales in the image. Liao et al. [14] designed a FCN-based TextBoxes model that directly output the coordinates of word bounding boxes at multiple network layers by jointly predicting text presence and coordinate offsets to a set of default regions on multiple layers. These kinds of methods treat text detection as object detection tasks. Instead of cropping the image to a number of sub-regions and evaluating the CNN thousands of times per image, these latest works regress the text region's coordinates on sparse proposals and detect text on a single forward pass. Our text detection system follows this strategy. To further boost the detection precision and maintain the high recall, a text relocation network is cascaded to estimate the accurate text regions on word level. It can also be used for refining the repeating detected text regions from multiple sliding windows or multiple channels.

III. Proposed method

In this section, we present the details of the proposed text detection method. The procedure of our system is shown in Fig. 2. It includes a fast text proposal extraction network operating in a high-recall and low-precision mode, and a refinement network aiming to relocate the overlapped text regions. The first part inputs the whole image and directly outputs regressed coordinates and text probability referring to a set of prior defined sparse windows. After thresholding on text probability in each window, thousands of non-text regions are removed. For overlapped text regions, the refinement network further estimates the accurate text locations on the relevant focused regions with a guidance mechanism. This system shows the effectiveness of exploring deep convolutional neural networks to directly output accurate text locations from text proposals to text, departing from previous approaches that apply a CNN classifier in a sliding window fashion requiring for a number of complicated post-processing steps.

TABLE I
ADDED LAYERS IN TEXT PROPOSAL GENERATION NETWORK.

Layer	Kernel size	Stride step	Nodes	Predict
Conv6	3	1	1024	Yes
Conv7	1	1	1024	Yes
Conv8_1	1	1	256	No
Conv8_2	3	2	512	Yes
Conv9_1	1	1	128	No
Conv9_2	3	2	256	Yes
Conv10_1	1	1	128	No
Conv10_2	3	2	256	Yes
Pool11	4	4	256	Yes

A. Object proposal-based text detection method

The text proposal generation network followes the idea in single shot multibox detecting (SSD) model [10]. It takes a natural scene image and fixed-size of windows to be classified as input and produces the adjusted word-region bounding boxes and the scores for the presence of text in them. After setting a threshold on the text probability, large amount of the regressed regions are removed. We call the rest of the detected regions text proposals. The architecture of this network is build on VGG-16 network but removes the fully-connected layers meanwhile adds auxiliary layers as described in Tab. II. Nine extra layers are concatenated after the basic VGG-16 network. On the top of several layers, side-output layers are expanded for sparse windows coordinates regression and text probability estimation. The Predict item in Tab. II represents whether the score and regional coordinates are predicted on that layer. Including Conv4_3 layer in VGG-16, seven layers output the prediction results for various text sizes. As for the prediction, it is performed in a convolutional fashion. 3×3 kernel size is used on each prediction layer to produce the score and a shape offset relative to the predefined sparse window's coordinates. In our system, we select six aspect ratios: 0.7, 1, 2, 3, 5, 7 for designing the sparse windows. The scales on the prediction layers range from 0.06 to 0.85. That means the lowest layer Conv3_3 has a scale of 0.06 and the highest layer Pool11 has a scale of 0.85, and all layers in between are regularly spaced. Since all the input images are normalized to 500×500, 38124 sparse windows are prepared and estimated in total. Most of the evaluated windows are non-text regions. Only the detections with text probability higher than 0.6 are remained as text proposals.

This method discredits the output space of sliding windows into a set of sparse windows over different aspect ratios and scales per feature map location. At prediction time, text score is estimated in each sparse window and produces adjustments to its coordinates to better match the word shape. Different windows may locate different parts of a word. So, it meets the same overlapping problem in sliding windows-based methods. Some overlapping results are shown in Fig. 3. The white score above the bounding box represents its text probability. It is difficult to remove the overlapped regions only by the score. Keep them as detection results decreases the precision. A relocation is performed on the overlapped text regions to get accurate text detection results.

Fig. 3. Examples of overlapped text detection results.

B. Text relocation with guidance

Inspired by the object relocation network LocNet [15] which aims to get more accurate detection regions of objects, we think out to apply it for text relocation network. Given a search region, the LocNet returns the bounding box of an object of interest inside this region. However, it only focuses on single object relocation. Computing the probability on each row and each column of the searched regions are independent. For the overlapped text region, it may contain multiple text lines. The vertical space between words in each text lines are different. So, computing the text probabilities on the column depends on the segmented text lines. Instead of using a linear structure to search the text lines firstly and then the word space in each text line through separated feature extraction steps, we propose a DNN-based text relocation network with guidance mechanism to extract features simultaneously. The text relocation network is shown in Fig. 4(a)). The guidance is a template to tell where to do max pooling over Y axis. Given a search space A, the CNN first extracts the features of it. Then max pooling transfers them on two directions and they are further input to two LSTM respectively. The output of LSTM is the vectors to estimate the text probabilities either of each row p_x=$p_x(i)_{i=1}^N$ or each column p_y=$p_y(j)_{j=1}^N$. The probability in each row or column represents the likelihood of this row or column belonging to the inside of text regions. After filtering out the rows and columns of low probability, the text bounding boxes can be achieved.

To remove the ambiguousness of searching words left and right boundaries among multiple text lines, the features extracted from CNN are first multiplied with the guidance before max pooling on Y axis. In training phase (Fig. 4(b)), the guidance is a given binary template designed from the highest ground truth text region. For example, if there are n text lines in the input image, the rows of the ground truth bounding box text regions are $\{[R_1, R_2], [R_3, R_4], \cdots, [R_{2n-1}, R_{2n}]\}$, the one with the maximum height is represented as $[R_t, R_b]$=$\{[R_{2x-1}, R_{2x}] | \max_x(R_{2x}\text{-}R_{2x-1})\}$. Then the template $B(i,j)$ is designed as Eq. 1.

$$B(i,j) = \begin{cases} 1, & R_t \le i \le R_b \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

In testing phase (Fig. 4(c)), the guidance is the binary template refereing to the rows of relocated text lines. We first compute the features from CNN and get the text probabilities of each row by the upper pipeline in Fig. 4(a). After

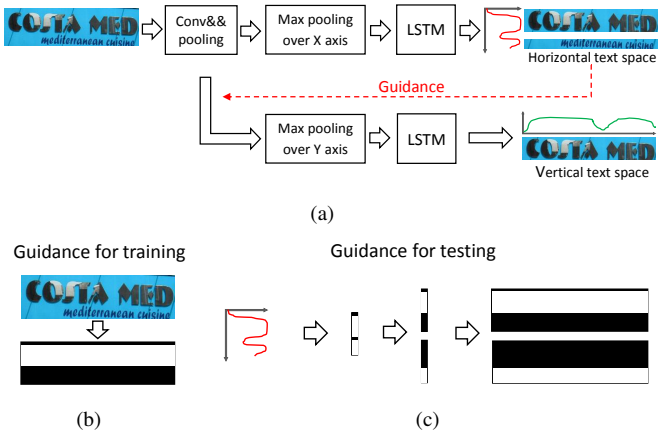


Fig. 4. Text relocation network with guidance mechanism. (a) The relocation network. (b) The guidance for training. (c) The guidance for testing.

thresholding on the text probabilities, the text lines can be obtained. Suppose there are m text lines $\{[RR_1, RR_2], [RR_3, RR_4], \cdots, [RR_{2m-1}, RR_{2m}]\}$, m corresponding templates $T_l(i,j)$ are generated from Eq. 2. So the process of max pooling and LSTM on Y axis are performed m times to get the spaces between words on m text lines.

$$T_l(i,j) = \begin{cases} 1, & RR_{2l-1} \leq i \leq RR_{2l}, l \in [1,m] \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

The feature extraction of text relocation network is implemented through VGG-16 network but removes the last pooling and fully-connected layers. The input image is normalized to 96 $\times$ 96. The output feature map size through the sequential convolutional layers becomes 6 $\times$ 6 $\times$ 512. Two max pooling layers pool the features to size 1 $\times$ 6 $\times$ 512 and 6 $\times$ 1 $\times$ 512 respectively. They are then input to LSTM and output two vectors of size 1 $\times$ 96 and 96 $\times$ 1.

For any two text regions, if their Intersection over Union (IoU) is larger than 0.2, they are considered as overlap. The relocation is performed on overlapped region iteratively until no new overlapping regions.

IV. Experimental Results

In this section, we introduce the databases and training methods for testing our method. The evaluation and analysis on scene text detection are presented as well.

Fig. 5. Examples of training samples for text relocation network.

A. Datasets

In the experiment, a Flicker image dataset [16] and the benchmark datasets SVT [17], ICDAR 2013 [18] are used for training and testing. In Flicker image dataset, text are generally attached on signboards and billboards with various font, color, orientation and position. Since our proposed method can only detect horizontal text, we exclude images in which text are not horizontal. In total, we get 2900 images from Flicker image dataset. All of those images and images in training set of SVT and ICDAR 2013 are used for learning. We test the model on testing set of ICDAR 2013 to evaluate the overall performance of our method.

B. Training

For text proposal generation network, only the input images and the ground truth of text regions are required in training phase. A relationship between each sparse window and the ground truth is built based on their IoU. The loss function is a weighted sum of the text localization loss and the text confidence loss [10]. We selected about 3452 training samples from Flicker image dataset, SVT training datasetand ICDAR dataset. To do data augmentation, we used the entire original input image and sub-regions of them. Three patches of the images are cropped so that the overlap with the text region is 0.5, 0.7, and 0.9. The size of each sampled patch is [0.5, 1] of the original image size.

For text relocation network, the input images with the corresponding ground truth vectors are required. Since the input are overlapped text regions, we extract those regions by extending the ground truth word regions on four directions (left, right, up and down). In total, 20,072 regions in Flicker image datasets are collected. The images and their ground truth are displayed in Fig. 5. The ground truth are binarized vectors representing the text regions in the row and word regions in one of the text line which has maximum height. GT_x(i)=1 means the i_{th} row belongs to text lines. Similarly, GT_y(j)=1 means the j_{th} column of a certain text line belongs to a word. The guidance is obtained from the vector of text lines. It should be a 2D template. To save the storage, we used a vector as input, and extend it to 2D when multiply with CNN features. The loss function is expressed in Eq. 3, where $\widetilde{p}$ = 1-p. p_x and p_y is the output of the network. With the input images and the ground truth vectors, this network can be trained end-to-end.

$$\sum_{s\in\{x,y\}} \sum_{i=1}^{N} GT_s(i) log(p_s(i)) + \widetilde{GT}_s(i) log(\widetilde{p_s(i)}) \quad (3)$$

C. Results and discussion

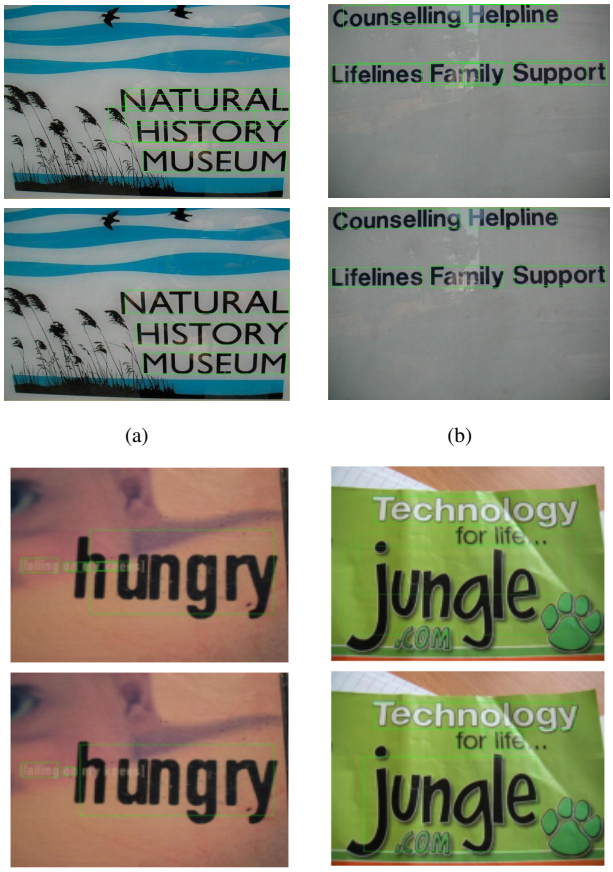
We test the relocation network on a collected testing dataset from Flicker images. The detail is shown in Fig. 6. First, the text probabilities of text lines are estimated (The spectrum and curve on left side of Fig. 6). For displaying, we normalize it to [0,255]. We set the thresholding to 0 to count the components. The number and range of the connected components (CCs) are also the numbers and range of the text lines in the image. In the

Fig. 6. The text relocation procedure. The spectrum and curve represents the text probabilities of each column and each row respectively.

(a)

(b)

Fig. 7. Text relocation results on Flicker images. (a) Simple scene text images. (b) Complicated scene text images.



(a) (b)

(c)

Fig. 8. Text relocation results on ICDAR 2013 images. The top ones are detected text regions by text proposal generation network. (a) Relocation results for vertical overlapped text. (b) Relocation results for horizontal overlapped text. (c) Failure cases after text relocation.

example, we detected two text lines. Then, based on the range of each text line, the inside regions of words are estimated in the same way. The number and ranges of CCs is the word number and ranges in that text line. Referring to the range of text line and the range of each word, it is easy to get the text bounding boxes. Many text relocation results are listed in Fig. 7. We can see that even the background of the images are complicated, text are adjoined to or being occluded by other objects, the relocation network can correctly predict the inside of text regions.

The proposed text detection model is tested on ICDAR 2013 for evaluating its text localization performance. Separately, we first test the performance of the first part, this is the text proposal generation network. Then, use the text relocation network for refinement. The results are summarized and compared with other methods in Tab. II. The results are evaluated under two different evaluation protocols, the DetEval [19] and the ICDAR 2013 evaluation [18]. Precision and recall and f-score are the measurement of detection performance. Precision represents the proportion of detected text regions to all detected regions. Recall is the proportion of detected text regions to ground truth text regions. f-score is a trade-off between precision and recall rate by computing their harmonic mean. From the comparison, we found that the precision improved a lot after the refinement of text relocation. Since the precision measures the detected text regions over all the detected regions, we can infer that after text relocation, lots of overlapped regions decrease and resulting in less false positive regions.

The text detection results are given in Fig. 8. The images on the top are the output of text proposal generation network and the bottom ones are the results after relocation. The text relocation network shows its super ability to predict the inside of text regions on both horizontal and vertical directions. However, this text relocation system fails in some cases. If the overlap IoU is smaller the setting value, the regions cannot be processed. Another case is shown in Fig. 8(c). If the text region is contained by another larger size text regions, it will be removed after relocation even it once detected in the former

TABLE II
COMPARISON OF DIFFERENT METHODS ON ICDAR 2013 DATASET.

Image dataset	ICDAR Eval			DetEval		
	Recall	Precision	f-score	Recall	Precision	f-score
Text proposal generation network	0.86	0.72	0.78	0.84	0.7441	0.79
Text proposal generation network + Text relocation network	0.84	0.83	0.84	0.86	0.83	0.85
CTPN [20]	0.93	0.83	0.88	-	-	-
TextBoxes [14]	0.88	0.83	0.85	0.89	0.83	0.86
SSD [10]	0.80	0.60	0.68	0.80	0.60	0.69
FCN [5]	0.88	0.78	0.83	-	-	-
FCRNall+filts [12]	-	-	-	0.92	0.76	0.83
TextFlow [21]	0.85	0.76	0.80	-	-	-

stage. Additionally, because it is difficult to define the ground truth of the top and bottom position for the overlapped text lines, this relocation system can only deal with horizontal text lines.

The proposed method is compared with other state-of-the-art text detection algorithms. The text proposal generation network is modified on SSD method which is generalized for object detection, but the performance improved greatly after we made it adaptive to text detection. Compared with other deep learning based methods, our text detection system with guided relocation is also comparable. In addition, the text relocation network can be combined with other regional-based methods for better text location performance.

V. CONCLUSION

This paper proposed a scene text relocation system that refined the detection from text proposals to text. It was structured by two cascaded deep neural networks, the text proposal generation network and text relocation network. The first network was a object proposal-based, and may produce text proposals that had overlap relation. To tackle this problem, a refinement DNN was applied on the overlapped regions through the procedure of feature extraction by CNN, max pooling over two directions and LSTM, to estimate text probability of each row and then each column with a guidance. This relocation network helped to remove the multiple overlapped text regions or joint cracked text regions. It showed the significance on precision boost after text relocation.

VI. ACKNOWLEDGMENTS

This project is partially supported by National High-tech R& Program of China(863 Program No.2015AA015403).

REFERENCES

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

[2] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and vision computing*, vol. 22, no. 10, pp. 761–767, 2004.

[3] W. Huang, Y. Qiao, and X. Tang, "Robust scene text detection with convolution neural network induced mser trees," in *Proceedings of the ECCV*, 2014, pp. 497–511.

[4] A. Zhu, R. Gao, and S. Uchida, "Could scene context be beneficial for scene text detection?" *Pattern Recognition*, vol. 58, pp. 204–215, 2016.

[5] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," *arXiv preprint arXiv:1604.04018*, 2016.

[6] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," 2016.

[7] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proceedings of the ICPR*, 2012, pp. 3304–3308.

[8] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *International Journal of Computer Vision*, vol. 116, no. 1, pp. 1–20, 2016.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "Ssd: Single shot multibox detector," *arXiv preprint arXiv:1512.02325*, 2015.

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

[12] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," *arXiv preprint arXiv:1604.06646*, 2016.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *arXiv preprint arXiv:1506.02640*, 2015.

[14] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," *arXiv preprint arXiv:1611.06779*, 2016.

[15] S. Gidaris and N. Komodakis, "Locnet: Improving localization accuracy for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 789–798.

[16] R. Gao, S. Uchida, A. Shahab, F. Shafait, and V. Frinken, "Visual saliency models for text detection in real world," *PloS one*, vol. 9, no. 12, p. e114539, 2014.

[17] K. Wang and S. Belongie, "Word spotting in the wild," in *European Conference on Computer Vision*. Springer, 2010, pp. 591–604.

[18] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. de las Heras, "Icdar 2013 robust reading competition," in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013, pp. 1484–1493.

[19] C. Wolf, E. Lombardi, J. Mille, O. Celiktutan, M. Jiu, E. Dogan, G. Eren, M. Baccouche, E. Dellandréa, C.-E. Bichot *et al.*, "Evaluation of video activity localizations integrating quality and quantity measurements," *Computer Vision and Image Understanding*, vol. 127, pp. 14–30, 2014.

[20] T. Zhi, H. Weilin, H. Tong, He an Pan, and Q. Yu, "Detecting text in natural image with connectionist text proposal network," in *European Conference on Computer Vision*. Springer International Publishing, 2016.

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