A Unified Deep Neural Network for Scene Text Detection

Yixin Li and Jinwen Ma(✉)

LMAM, Department of Information Science, School of Mathematical Sciences,
Peking University, Beijing 100871, China
liyixin@spku.edu.cn, jwma@math.pku.edu.cn

Abstract. Scene text detection is important and valuable for text recognition in natural scenes, but it is still a very challenging problem. In this paper, we propose a unified deep neural network for scene text detection, which is composed of a Fully Convolutional Network (FCN) for text saliency map generation and a Bounding box Regression Network (BRN) for text bounding boxes prediction. The FCN is trained with a hybrid loss function based on two types of pixel-wise ground truth masks while the unified neural network is fine-tuned with a multi-task loss function. Additionally, the post-processing procedures including scoring the predicted bounding boxes by the saliency map and eliminating the redundant boxes via the Non-Maximum Suppression (NMS) method are applied to improve the final text detection results. It is demonstrated by the experimental results on ICDAR2013 benchmark that our proposed unified deep neural network can achieve good performance of text detection and process images at 5 fps, being faster than most of the existing text detection methods.

Keywords: Scene text detection · Fully Convolutional Network · Deep learning

1 Introduction

As a series of abstract symbols for human communication, text carries rich contextual and semantic information. Reading and understanding text in a natural image plays an important role in many computer vision tasks, such as image matching, robot navigation and human-computer interaction. Since the background can strongly affect the result of text recognition, it is necessary to detect or localize text lines in natural images before text recognition. Thus, text detection has become a popular research topic in text recognition and computer vision.

Although Optical Character Recognition (OCR) is considered to be a powerful character recognition tool for the scanned images, it is still rather difficult to detect and recognize text in natural scenes. In fact, we can spot and localize text instantly just by a glance of the scene, even if the text is written in a language we do not know. However, real-time text detection in natural scene can be a difficult task for a computer, due to the diversity of scene text, complexity of background and low quality of natural images [1]. Actually, text can be in different colors, fonts, sizes, and orientations even in a single natural image. On the other hand, some common objects in natural scene with certain textures like fences, trees and traffic signs, can be easily confused with text, and produce

D.-S. Huang et al. (Eds.): ICIC 2017, Part I, LNCS 10361, pp. 101–112, 2017.
DOI: 10.1007/978-3-319-63309-1_10

false positive samples to the text detection system. Moreover, natural images with low resolution, non-uniform illumination, and partial occlusion are not uncommon, which can be another challenge to the detection system.

To tackle these difficulties, a variety of methods and algorithms have been established from different aspects. The primary focus of these approaches is generally to learn an effective and robust feature representation of text. It is an extremely difficult task, due to the variation of text and interference from the environment. Conventional methods treat text as regions with certain texture, which are sensitive to the uneven lighting conditions and background interferences causing by human defined rules and handcraft features. Deep learning based methods are more robust under the supervised training on large amounts of labeled data. Most of the deep learning based methods predict a saliency map so that they are unable to generate bounding boxes directly. In this case a post-processing algorithm is needed to generate the final detection results.

In this paper, we propose a deep learning architecture for scene text detection which is composed of a Fully Convolutional Network (FCN) [2] and a Bounding Box Regression Network (BRN). The unified deep neural network is trained to generate a pixel-wise saliency map and predict the locations of candidate bounding boxes simultaneously. Then we score the generated bounding boxes by the text saliency map, and the non-maximum suppression (NMS) method [3] is adopted to filter the overlapping bounding boxes. It is demonstrated by the experiments on ICDAR2013 dataset [4] that our proposed unified network can achieve good text detection performance. Moreover, it can process 5 images per second, being faster than most of the existing scene text detection methods.

The rest of this paper is organized as follows. We briefly review the related text detection methods in Sect. 2. The methodology of our scene text detection approach is introduced in Sect. 3. Section 4 summarizes the experiment results and comparisons. Finally, a brief conclusion is given in Sect. 5.

2 Related Works

In recent years, with more and more attention to text detection, many effective algorithms and strategies have been established to extract and locate text in natural images. Most of current text detection methods mostly have a step-wise pipeline. They extract letter or word candidates or generate a text saliency map from the input image, then group the word candidates or regions with high response in saliency map into text lines, some of the approaches filter the text lines using an offline trained classifier to achieve higher precision.

Epshtein et al. [5] proposed the stroke width transform (SWT) to extract letter candidates and then merge them into text lines. Maximally Stable Extremal Regions (MSER) [6] was proposed in the early 2000s as a kind of affine invariant regions, and was brought into text detection task in 2010s [7]. MSER-based text detection algorithms [7, 8], taking MSERs as letter candidates, achieved stat-of-the-art performance on ICDAR2013 [4] benchmark, USTB_STAR [9] in particular, even won the ICDAR2013 robust reading competition [4]. Sun et al. [10] proposed the Color-enhanced Extremal Regions (CER)

method which has been one of the most powerful text detection approaches without any deep learning technology. But these kinds of methods extract letter candidates with human defined rules and filter them with handcraft features, followed by many parameters which are very difficult to optimize.

In the past few years, deep learning, especially convolution neural networks, has been widely used in almost all the computer vision tasks, and text detection is no exception. Because of the powerful generalization abilities and the supervised training on big data, deep learning based text detection methods beat the conventional methods to the top of benchmarks. Also, the parameters in deep neural networks can be learned automatically. In the Text-spotter [11], a CNN filter was utilized to perform a sliding window type search and produce a saliency map to predict text regions. Zhang [12] and Yao [13] considered text detection as a semantic segmentation task, applied a FCN to generate a pixel-wise saliency map and then utilized a graph partition or classification algorithm to localize text lines. Although the deep neural networks can perform a feed-forward procedure rather fast after the training phase is completed, these methods can hardly be real-time due to the complex post-processing procedures.

3 Methodology

3.1 Overview

In this section, we describe our unified deep learning neural network architecture for scene text detection in detail. As shown in Fig. 1, our proposed text detection pipeline has two parts: a single deep neural network that can simultaneously generates text saliency map and locates the candidate bounding boxes, and an extremely simple post-processing procedure that filters the overlapping bounding boxes.

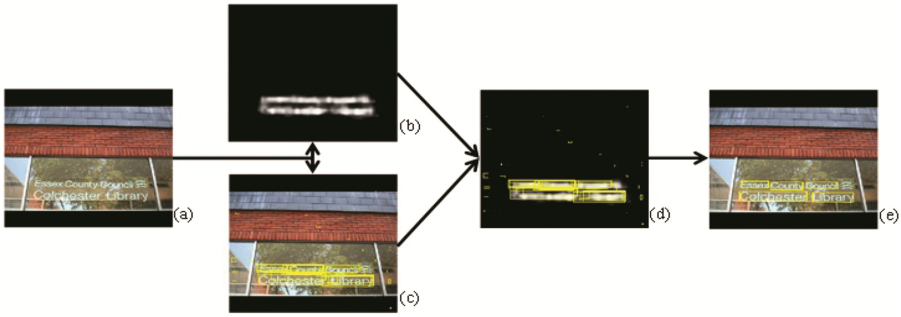

Fig. 1. The pipeline of the proposed unified text detection method. (a) Original image; (b), (c) Text saliency map and bounding boxes generated by the network; (d) Candidate bounding boxes on text saliency map; € Final detection result.

For the network part, we respectively train two networks: a Fully Convolutional Network (FCN) to predict each pixel and a small sized network sharing feature with the FCN called bounding box regression network (BRN) to learn the locations of bounding

boxes. Then, these two networks sharing the layers before 'pool5' layer of the FCN are fine-tuned with a multi-task loss function together. The architecture of our unified deep neural network is shown in Fig. 2.

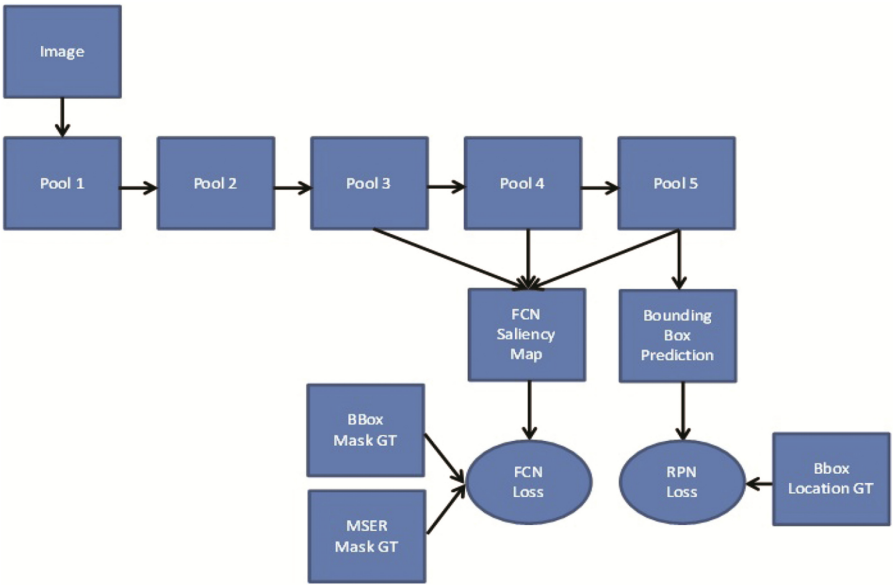

Fig. 2. The architecture of our proposed unified deep neural network.

As for the post-processing part, we use the saliency map generated by the FCN to score the candidate bounding boxes predicted by the BRN. Then, the non-maximum suppression (NMS) method is adopted to filter the overlapping bounding boxes.

In the time of test, we feed images into the unified deep neural network and get a text saliency map and bounding box predictions simultaneously. Then, each candidate bounding box is scored by the saliency map. At the post-processing stage, the NMS method is applied to eliminate the redundant bounding boxes to get the final text detection results.

3.2 Fully Convolutional Network (FCN)

Sliding window type methods, like the Text-spotter [11], generate text saliency map by classification confidence of each window. Each value of the confidence map only considers the feature within a limited region. Some objects like tree leaves, traffic signs and piano keyboards, with parts very similar to text, can be easily misclassified to text and produce false positives.

Actually, Fully Convolutional Network (FCN) [2] was recently proposed to solve the semantic segmentation problem, and achieved state-of-the-art performance on PASCAL VOC dataset. By the means of deconvolution and upsampling, different feature layers from a classification network are able to merge into one single feature

map and perform a pixel-wise prediction. Since FCN uses different feature layers including shallow layers with low level feature and deep layers that carry high level semantic information, it considers both local and global information, thus it can 'see the bigger picture' then a sliding window type neural network.

Inspired by the idea of text detection via semantic segmentation given in [12, 13], we use FCN to classify each pixel to text or non-text. We modify VGG-16 [14] to our FCN architecture by removing the fully connected layers and adding deconvolution layers after pooling stages. In our FCN architecture, the 3rd, 4th and 5th pooling layers are fused into one feature map by adding deconvolution layers and upsampling by 8,16 and 32 times using bilinear interpolation, resulting in the prediction feature map with the same size of input image. The FCN is trained on a log softmax loss function to perform the pixel-wise prediction.

In order to train the FCN, we need a pixel-wise image label for each training image, but most available text detection datasets [4, 15, 16] provide ground truth by bounding box parameters. The most straightforward way is to set pixels within ground truth bounding boxes to positive and pixels outside text bounding boxes to negative like the second row of Fig. 3. But this may lead to a so called 'sticking' problem [13]: when multiple text lines are close to each other, we may not be able to separate them from the saliency map predicted by FCN.

Fig. 3. Two types of ground truth mask. Original images are in the first row, the second and third rows contain bounding box ground truth masks and MSER ground truth masks.

To tackle this problem, we utilize the Maximally Stable Extremal Region (MSER) [6] ground truth mask and we train the FCN with a hybrid loss function. Since text components carry rich edge information and generally have significant color contrast with backgrounds, many text detection approaches (e.g. [7, 9, 10, 17]) extract the MSERs or Ers as letter candidates. In fact, we extract the MSERs within the bounding box ground truth, and only set the pixels in the MSERs as positive. As shown in the third row of Fig. 3, the contour and shape of letters are well described by the MSER ground truth mask. Thus, we can define the hybrid loss function as follows:

$$\mathcal{L}_{FCN}(M, M_{bbox}, M_{mser}) = \lambda_{bbox} \cdot \mathcal{L}_{\log softmax}(M, M_{bbox}) + \lambda_{mser} \cdot \mathcal{L}_{\log\ softmax}(M, M_{mser}), \quad (1)$$

where M is the output prediction map of the FCN, and M_{bbox} and M_{mser} are the bounding box and MSER ground truth map, respectively. Accordingly, the FCN can learn the region of text lines by minimizing the first part of loss function, and learn the contour and shape of letter by minimizing the last part of loss function. In our experiments, we treat the loss functions of two types of ground truth mask equally, thus the parameters λ_{bbox} and λ_{mser} are both set to 0.5.

We modify the VGG-16 pre-trained on the ImageNet dataset to our FCN model, train the FCN by minimizing the hybrid loss function defined in (1) by the stochastic gradient descent. During the training phase, we inherit the most of the used parameters including the learning rate with 1e−4, the weight decay with 0.9 and the dropout with 0.5 from the FCN model proposed in [2].

3.3 Bounding Box Regression Network (BRN)

R-CNN based object detection approaches treat the detection as a classification problem. A region proposal algorithm, selective search (SS) or deep neural network for instance, can be utilized to generate candidate bounding boxes, and then a classifier is adopted to score each bounding box. Finally, the bounding boxes are filtered according to their confidence scores and the remaining bounding boxes become the final detection result. This kind of framework is inefficient and hard to optimize because the great number of candidate bounding boxes causes a mass of redundant calculation and each part of the framework must be trained individually. The YOLO (You Only Look Once) approach [18] treats the detection as a regression problem directly and a unified neural network can be trained to learn the locations of bounding boxes directly. Due to this simple structure, the YOLO approach can run in real-time, and in the meantime, achieve a high detection performance.

Our main idea is to train a regression network to learn the locations of text bounding boxes directly. This regression network we called bounding box regression network (BRN) shares the same layers before 'pool5' layer as the FCN and a small sized neural network is added to perform the regression. The BRN can be fine-tuned with the FCN together.

Instead of using fully connected layers in the YOLO structure, we use convolution layers in BRN to maintain the spatial information. Inspired by Darknet-19 [19], we apply 1×1 convolution layers to reduce the number of the feature map tunnels, and batch normalization layers to accelerate the convergence and avoid the overfitting.

Like the YOLO model, our proposed BRN can be regarded as dividing $32n \times 32n$ input image to $n \times n$ even grid. If a text bounding box region overlaps with a grid cell, then this cell is responsible for detecting the nearest text bounding box. The YOLO network learns 4 location coordinates and a confidence score for each bounding box in a grid cell, and learns C class probabilities. Since we just have one class of text, it is unnecessary to learn the class probability. So, we only train BRN to learn 4 location

parameters, and the confidence score can be calculated from the FCN saliency map. Thus, our BRN output map is $12 \times 12 \times 4$ (for a 384×384 image).

The four location coordinates are relative distance from four bounding box edges to the center of the grid cell. We use a Sigmoid function to convert the coordinates to 0 to 1. The L2 loss we use weights each grid cell equally, but we want grid cells that overlap with text bounding box to be more important and outweigh the cells with no text region in it. Therefore, we introduce two parameters to control the weights between text cells and non-text cells. The loss function is designed as follows:

$$\begin{aligned}\mathcal{L}_{BRN}(C, \widehat{C}) = \lambda_{text} \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} I_{text}^{ij} \left\| C_{ij} - \widehat{C}_{ij} \right\|_{L^2}^2 \\ + \lambda_{non\text{-}text} \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} (1 - I_{text}^{ij}) \left\| C_{ij} - \widehat{C}_{ij} \right\|_{L^2}^2\end{aligned} \tag{2}$$

where C and $\widehat{C}$ are the output of BRN and the coordinates ground truth, I_{text}^{ij} denotes if a text bounding box region overlaps with the grid cell ij. And the two parameters λ_{text} and $\lambda_{non-text}$ are set to 10 and 1 in our experiments. And we set the learning rate to 1e−5 to avoid gradient explosion.

3.4 Joint Training of FCN and BRN

Our idea is to generate text saliency map and predict the locations of text bounding boxes in the same time by a single deep neural network. So we dock the FCN and BRN by connecting the 'pool5' layer and sharing the feature layers before. We then fine-tune the whole unified network architecture by optimizing a multi-task loss function combined with the loss functions of FCN and BRN:

$$\begin{aligned}&\mathcal{L}(M, M_{bbox}, M_{mser}, C, \widehat{C}) \\ &\quad = \lambda_{FCN} \cdot \mathcal{L}_{FCN}(M, M_{bbox}, M_{mser}) + \lambda_{BRN} \cdot \mathcal{L}_{BRN}(C, \widehat{C})\end{aligned} \tag{3}$$

In our experiments, we set λ_{FCN} and λ_{BRN} to be 10 and 1, respectively. We gradually reduce the learning rate from 1e−4 to 1e−5 to avoid the gradient explosion.

3.5 Non-Maximum Suppression (NMS)

Our BRN output map divides the input image into even grid cells which are 32×32 patches of the input image. That is, the BRN predicts one bounding box for every grid cell, but most of them are redundant. For instance, the BRN generates a $12 \times 12 \times 4$ map for a 384×384 input image, which is 144 bounding boxes predicted by the BRN. We score these bounding boxes by the saliency map from the FCN, and then apply the non-maximum suppression (NMS) algorithm to eliminate the redundant bounding boxes and get the final detection result.

In the scoring procedure, we would like the bounding box with more text content and less non-text content gets a higher score. That is, the bounding box should contain as much pixel value of saliency map as possible. Thus, we define the confidence score as follows:

$$score(R) = \sum_{(i,j)\in R} map_{ij},$$

where map_{ij} is the pixel-wise value of text saliency map generated by the FCN.

Then, we apply the non-maximum suppression (NMS) algorithm to filter the redundant bounding boxes by their confidence scores. That is, if the overlap ratio of two bounding boxes is higher than a given threshold, we remove the one with lower confidence score. The post-processing procedure only takes about 0.03 s on a 3.0 Hz CPU.

4 Experiments

In this section, we begin to introduce the datasets and present the experiment results and comparisons of our proposed method on them. Moreover, the running time and limitations of our proposed method are also discussed.

4.1 Datasets

All the images used to train our network are harvested from ICDAR2013 [4], ICDAR2015 [15], and COCO-Text [16] with data augmentation.

ICDAR2013. The 'robust reading' competition held by International Conference on Document Analysis and Recognition (ICDAR) every two years provides a dataset and a benchmark on scene text detection. In fact, ICDAR2013 [4] contains about 500 images with text annotations for training and testing. We apply flip and rotation to make our network capable to detect multi-orientated text layout.

ICDAR2015. ICDAR2015 'robust reading' competition [15] is divided into four different tasks. While task 2 'Focused Scene Text' is same as ICDAR2013 which is well-captured by camera and focus on horizontal text lines, task 4 'Incidental Scene Text' is introduced as a new task. This dataset is captured by wearable device without prior knowledge of the whereabouts of text lines. ICDAR2015 dataset is more close to the real word scenario so it is much more difficult than the previous ICDAR2013 dataset. We apply random crop to augment the data.

COCO-Text. COCO (Common objects in content) is a huge image dataset for numerous computer vision tasks, such as object detection, semantic segmentation and human key point detection. With text instance labeling on COCO dataset, it can be used for text detection task, and this subset of COCO is called COCO-Text [16]. We filter the text instances and remove the ones with low quality and we take 12172 images for training and 5641 images for validation.

4.2 FCN Predictions

We train three FCNs by different kinds of ground truth mask individually to evaluate the performance on pixel-wise predictions. We call these three FCNs trained by bounding box ground truth mask, MSER ground truth mask, and hybrid ground truth mask using the loss function defined by (1) BboxNet, MserNet and HybridNet.

All three FCNs can predict the text lines in the most scenarios as shown in the first row of Fig. 4. But when two text lines are very close, the prediction of BboxNet tends to be a blob of high response which makes us unable to separate the text lines. For example, the 'Key' and 'West' cannot be separated from the BboxNet saliency map in the second row of Fig. 4. There are still some problems when we train the FCN only by MSER ground truth mask. The most severe one is that, when the stroke width of the text is relatively large and the text fills up the nearby region, the MserNet might sometimes get the background and text regions mixed up like the 'animal' in the last row of Fig. 4. While the HybridNet is capable of separating the close text lines, and in the meantime, it is unlikely to be confused by the background and text regions.

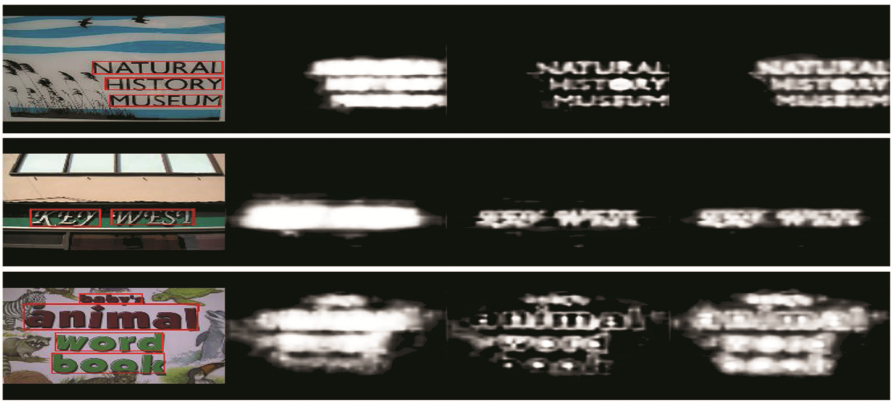

Fig. 4. Saliency maps produced by BboxNet, MserNet and HybridNet.

4.3 Experiment Results

The BRN is trained to learn four parameters of horizontal bounding boxes. So we test our proposed text detection system on ICDAR2013 dataset and pass the ICDAR2015 and COCO-Text which has images with multi-oriented text instances. Our method achieves 0.79 precision, 0.73 recall and 0.76 F-measure on ICDAR2013 benchmark. The evaluation details can be found in the official document of ICDAR2013 robust reading competition [4].

The most of the top methods on the ICDAR2013 leaderboard have not been described in academic papers to date, so we only list our proposed method along with other methods which are published in Table 1.

Table 1. Test results on ICDAR2013 dataset

Methods	Precision	Recall	F-measure	Time/s
CER [10]	0.87	0.86	0.86	
FCN_Megvii [13]	0.89	0.80	0.84	0.62
Proposed	0.79	0.73	0.76	0.19
USTB_TextStar [9]	0.88	0.66	0.76	0.80
Text-Spotter [11]	0.88	0.65	0.74	>1[a]
CASIA [8]	0.85	0.63	0.72	
I2R NUS [4]	0.73	0.66	0.69	
TH-TextLoc [4]	0.70	0.65	0.67	
TD-Mixture [20]	0.69	0.66	0.67	7.20
SWT [5]	0.73	0.60	0.66	>3[a]

[a]Evaluated by our experiments

It can be found that our precision is much lower than most of other methods, since our proposed network treats the detection as a regression problem directly without region proposals, the locations of predicted bounding boxes might be inaccurate which leads to a low precision performance. On the bright side, our proposed method is extremely fast for no redundant calculation from region proposals.

The detection samples in Fig. 5 show that our method is able to handle text lines in different colors, fonts and scales and performs well in several scenarios.

Fig. 5. Detection samples of the proposed unified deep learning neural network.

4.4 Running Time

The framework of the proposed method is very simple. On a 384 × 384 image, it takes only 0.12 s on average to generate a text saliency map and bounding box predictions running on a single GTX1080 GPU without batch input. Actually, it should be at least twice faster when running on a GTX Titan X GPU. The post-processing procedure of NMS can be efficiently done on a single CPU in 0.03 s. Our whole text detection system

can process 5 images per second but it is still far from real-time. The deep learning platform we use in our experiments is MatConvNet [21].

4.5 Limitations

Since the FCN is sensitive to several certain scenarios including uneven illumination and blur, our proposed text detection system might fail under these certain conditions. Light spots causing by the reflection of light source split the text when they appear in the middle of text lines. Moreover, the severe blur on text regions makes the FCN difficult to extract the feature representation from the text regions.

Our unified deep neural network directly learns the locations of text bounding boxes as a regression problem. Therefore, the bounding boxes predicted by the BRN are not so accurate in comparison with the other classification based methods, which leads to a decrease on precision. Moreover, we only train the BRN to learn horizontal rectangle bounding boxes for text lines, but text in real life images can be in different layouts like multiple orientations or circles. Our system might fail when it comes to these scenarios.

5 Conclusion

We have established a unified deep neural network architecture for scene text detection. A Fully Convolutional Network (FCN) is trained to predict the text saliency map in a pixel-wise style with a hybrid loss function to overcome the 'sticking' problem. Moreover, the Bounding box Regression Network (BRN) sharing the feature layers with the FCN is directly trained with the locations of indexed bounding boxes. The unified network is fine-tuned in an end-to-end manner by a multi-task loss function. For scene text detection, we input the natural images to the unified network to generate a text saliency map and predict the locations of candidate bounding boxes at the meantime. We then score each bounding box by the saliency map and use the non-maximum suppression (NMS) algorithm to eliminate the redundant bounding boxes. It is demonstrated by the experimental results on ICDAR2013 benchmark that our unified network can achieve 0.76 F-measure and run at 5 fps on a GPU, which is faster than most of the other existing text detection methods.

Acknowledgements. This work was supported by the Natural Science Foundation of China for Grant 61171138.

References

1. Zhu, Y., Yao, C., Bai, X.: Scene text detection and recognition: recent advances and future trends. Front. Comput. Sci. **10**(1), 19–36 (2016)
2. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **PP**(99), 640–651 (2014)
3. Neubeck, A., Gool, L.V.: Efficient non-maximum suppression. In: International Conference on Pattern Recognition, pp. 850–855. DBLP (2006)

4. Karatzas, D., Shafait, F., Uchida, S., et al.: ICDAR 2013 robust reading competition. In: International Conference on Document Analysis and Recognition, pp. 1484–1493. IEEE Computer Society (2013)
5. Epshtein, B., Ofek, E., Wexler, Y.: Detecting text in natural scenes with stroke width transform. In: Computer Vision and Pattern Recognition, pp. 2963–2970. IEEE (2010)
6. Matas, J., Chum, O., Urban, M., et al.: Robust wide-baseline stereo from maximally stable extremal regions. Image Vis. Comput. **22**(10), 761–767 (2004)
7. Neumann, L., Matas, J.: A method for text localization and recognition in real-world images. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010. LNCS, vol. 6494, pp. 770–783. Springer, Heidelberg (2011). doi:10.1007/978-3-642-19318-7_60
8. Shi, C., Wang, C., Xiao, B., et al.: Scene text detection using graph model built upon maximally stable extremal regions. Pattern Recogn. Lett. **34**(2), 107–116 (2013)
9. Yin, X.C., Yin, X., Huang, K., et al.: Robust text detection in natural scene images. IEEE Trans. Pattern Anal. Mach. Intell. **36**(5), 970–983 (2014)
10. Sun, L., Huo, Q., Jia, W., et al.: A robust approach for text detection from natural scene images. Pattern Recogn. **48**(9), 2906–2920 (2015)
11. Jaderberg, M., Vedaldi, A., Zisserman, A.: Deep features for text spotting. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8692, pp. 512–528. Springer, Cham (2014). doi:10.1007/978-3-319-10593-2_34
12. Zhang, Z., Zhang, C., Shen, W., et al.: Multi-oriented text detection with fully convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4159–4167 (2016)
13. Yao, C., Bai, X., Sang, N., et al.: Scene text detection via holistic, multi-channel prediction. arXiv preprint arXiv:1606.09002 (2016)
14. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
15. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., et al.: ICDAR 2015 competition on robust reading. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 1156–1160. IEEE (2015)
16. Veit, A., Matera, T., Neumann, L., et al.: Coco-text: dataset and benchmark for text detection and recognition in natural images. arXiv preprint arXiv:1601.07140 (2016)
17. Huang, W., Qiao, Yu., Tang, X.: Robust scene text detection with convolution neural network induced MSER trees. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8692, pp. 497–511. Springer, Cham (2014). doi:10.1007/978-3-319-10593-2_33
18. Redmon, J., Divvala, S., Girshick, R., et al.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
19. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. arXiv preprint arXiv:1612.08242 (2016)
20. Yao, C., Bai, X., Liu, W., et al.: Detecting texts of arbitrary orientations in natural images. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1083–1090. IEEE (2012)
21. Vedaldi, A., Lenc, K.: Matconvnet: convolutional neural networks for matlab. In: Proceedings of the 23rd ACM International Conference on Multimedia, pp. 689–692. ACM (2015)