Scene Text-Line Extraction with Fully Convolutional Network and Refined Proposals

Guan-Xin Zeng, Yu-Hong Hou, Po-Chyi Su*, Li-Wei Kang†
*Dept. of Computer Science and Information Engineering, National Central University, Taiwan
E-mail: pochyisu@csie.ncu.edu.tw Tel: +886-3-4227151x35314
† Department of Electrical Engineering, National Taiwan Normal University, Taiwan,
E-mail: lwkang@ntnu.edu.tw Tel: +886-2-7749-3563

***Abstract*— Texts appearing in images are often regions of interest. Locating such areas for further analysis can help to extract image-related information and facilitate many applications. Pixel-based segmentation and region-based object classification are two methodologies for identifying text areas in images and have their own pros and cons. In this research, a text detection scheme consisting of a pixel-based classification network and a supplemented region proposal network is proposed. The main network is a Fully Convolutional Network (FCN) employing Feature Pyramid Networks (FPN) and Atrous Spatial Pyramid Pooling (ASPP) to indicate possible text areas and text borders with high recall. Certain areas are further processed by the refinement network, i.e., a simplified Connectionist Text Proposal Network (CTPN) with high precision. Non-Maximum Suppression (NMS) is then applied to form appropriate text-lines. The experimental results show feasibility of the scheme.**

I. INTRODUCTION

Texts appearing in images always convey plentiful information. These identifiable markers such as traffic signs, shop signs, posters and slogans, etc. usually draw a lot of attention and can thus be viewed as regions of interest in images. Locating the corresponding areas in streetscape may help to extract the image-related information, such as the locations where the pictures were taken, or to evaluate the effects of advertising billboards. This research aims at detecting texts in images. It should be noted that text detection in such images as streetscape is challenging because of relatively complex content. For example, contours of buildings, roads or trees exist and store/road signs that may overlap each other would further complicate the corresponding detection or recognition. With the rapid progress of deep learning technologies, many methods of text detection have been developed. Deep learning methods for text detection mainly have two methodologies as shown in Fig. 1, i.e., pixel-based segmentation and region-based object detection. Both methodologies have their own pros and cons so, in our opinions, making good use of the properties of these two may help to achieve better results. In this research, we designed a network architecture based on a semantic segmentation network, which is supplemented by an object detection network. In order to deal with texts with large differences in size and aspect ratio, we combine Feature Pyramid Networks (FPN)[1] and Atrous Spatial Pyramid Pooling (ASPPP)[2] into the segmentation process. No complicated post-processing is involved but effective fusion by Non-Maximum Suppression (NMS). The rest of the paper is organized as follows. The proposed scheme will be detailed in Sec. 2, followed by the test results on ICDAR2013[3] dataset in Sec. 3. Conclusion and future work will be given in Sec. 4.

Fig. 1 Pixel-based methods tend to identify all the pixels covering texts while region-based methods find a bounding box enclosing texts.

II. THE PROPOSED SCHEME

Fig. 2 shows an overview of the proposed scheme, which consists of a main network based on pixel-based classification and a refined network based on a region based classification method. The next subsections will describe the three major steps of the proposed scheme.

A. Main Network

The structure of the main network is illustrated in Fig 3. The main network employs the ResNet-101[4] as the backbone of feature extraction. By removing the last layer of ResNet block, we extract the feature map of the fourth layer, which is 1/16 the size of the original one to carry out more intensive feature extraction. In addition, we also extract the feature map of the second layer from the ResNet-101 backbone network, which is a quarter of the original size and will be used for the latter half of the main network. ASPP layer contains one 1x1 convolution, three Atrous convolution with the size 3x3 and the expansion rate of {6,12,18}. All of them contain the batch normalization and image-level features for global average pooling. Then we use 1x1 convolution mixed features to acquire larger receptive fields. Each Atrous convolution output has a fixed depth 256. The depthwise separable convolution is used as much as possible to reduce the amount of calculation. Finally, all the results are

concatenated into decoder features and compressed back to 256 channel using 1x1 convolution kernel, which can make each layer of features fuse with each other.

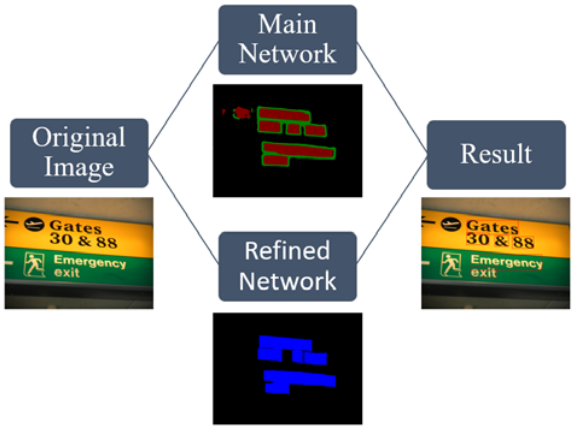

Fig. 2 The flowchart of the proposed scheme bounding box enclosing texts

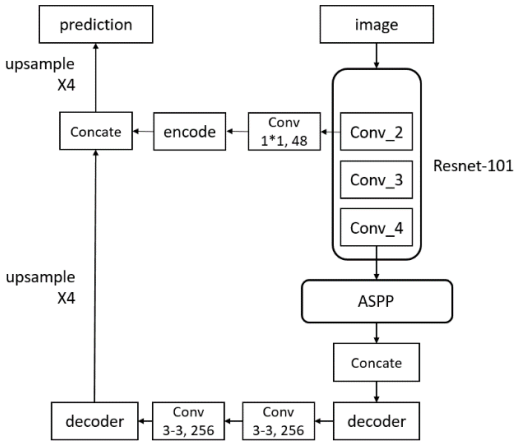

Fig. 3 Main network structure

The decoder features are up-sampled 4 times from 1/16 the original image size to 1/4 by bilinear interpolation. Then, we extract Conv2_x feature map with the size equal to 1/4 the original image from Resnet-101, and compress it with 1x1 convolution into encoder feature with a depth of 48. The encoder feature with the low-level features and decoder features are concatenated and mixed with two 3x3 convolution kernels. The final output is a decoder feature with a depth of 256. The decoded feature map maintains high spatial resolution and semantic information. Finally, a 1×1 convolution kernel is used to output a convolution layer with the number of categories, specifically three in this case, to implement the class prediction. The classification result is scaled back to the original size with bilinear interpolation.

Common semantic segmentation tasks such as CamVid[5] are usually multi-category classifications while the considered case here is a three-category (text, text edge, and background) problem. When the number of categories is limited, object imbalance may exist and different weighting factors should be assigned for each category. We try two different text labeling methods as illustrated in Fig. 4. The first method is an intuitive way by adding borders around the text areas. The second method is to shrink the text slightly, which is expected to reduce errors of manual labeling and highlight the characteristics of texts. Fig. 5 shows the effect of training in these two labeling methods. The left panel of the figure shows the results without shrinking so the edges are thinner and could be unstable. The right panel shows the results with shrinking texts and the edges can be drawn more completely. Therefore, we chose to adopt the shrinking text label in the proposed scheme.

Fig. 4 Two different labeling approaches

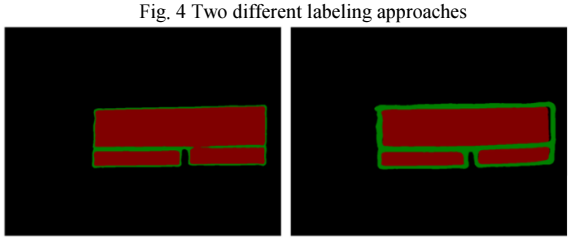

Fig. 5 An example of classification results of different labeling methods

B. Region-based refined network design

For the refined network, we tend not to choose a complex anchor design architecture, because testing with many and diverse anchors is cumbersome and time-consuming. Connectionist Text Proposal Network (CTPN) [6] is considered in our design. In CTPN, a lot of small bars are used to detect texts. These bars are evaluated and connected to form text-lines. Since we don't need a very time-consuming merging algorithm, lightweight CTPN with improved cut-off should be a good choice here.

The overview of the refined network is illustrated in Fig. 6. The refined network is divided into two steps. We use Resnet-101 as the backbone network to extract the feature. The results of the third layer (conv3_x) and the fourth layer (conv4_x) are extracted, and the feature map sizes are 1/8 and 1/16 of the original image size respectively. Then we use the 1x1 convolution kernel to compress the channel to P3 and P4 respectively so that the depth is the same as 256. P4 uses the bilinear interpolation to up-sample to P3, and concatenate the two layers into “rpn_conv”. We further use two 3x3 convolution kernels to mix the features on “rpn_conv”. The advantage is that we can make better use of the performance of the feature networks and use the lower-layer feature maps to achieve better positioning accuracy. It also avoids the loss of smaller words at higher levels and helps to keep the rich semantic features of high-level feature maps. Next, we use the RPN network layer to find the text blocks. Because our

feature map is 8 times different from the original image, we use an anchor with an equal width of 8. Fourteen different heights from 6 to 254 are tested. These anchors help to locate texts on the feature map and provide each box a "text score" with two regression parameters, i.e. y coordinates and heights.

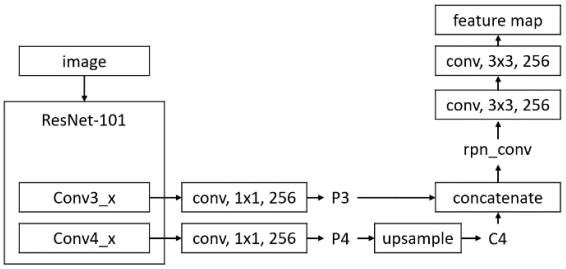

Fig. 6 Refined network feature extraction

C. *Post-processing*

When an input image passes through our main network and the refined network, we obtain two separate results. Combining these results to acquire a better outcome is the objective of this step. In addition, it is preferred not to increase too much computational burden in this post-processing. There are basically two steps; in Step 1, the result of the main network is detected by the outline of the green area (edge). Only the part with complete inner contour is retained. The inner contour means that it is a hollow area indicating the text area marked with red is what we need. Then we check each point of the inner contour to see if there is a red area nearby. If it is confirmed that there is a completely covered red area, the area is considered a text area and the red area is framed as the minimum external rectangle. In Step 2, the remaining red areas will be evaluated by the results of the refined network. If the refined network also predicts it as a text area, then it will be ruled as containing texts. Otherwise, it is considered a wrong detection. In Step 3, we collect the results of the first and the second batches. If we can find some small sporadic boxes that mainly occur in the red blocks mixed in the green area, they are usually covered by a large frame. We then designed an NMS to filter and acquire the final results.

III. EXPERIMENTAL RESULTS

A. *Main network detection results*

Fig. 7 shows an example of our main network detection. The results of the main network can be explained in two directions: 1) Only the red areas surrounded by the green areas (edge) and 2) All the red areas. We used the ICDAR2013 test set to examine the difference between them, with the scores (excluding the refined network) shown in Table I. We first observe that Table I (B) shows a very high recall rate but a lower precision. This means that the model has identified most of the texts, but quite a few errors also happen. Let's take a look at the Table I (A). After only the red areas surrounded by the green area are framed, the precision increases significantly and the recall decreases slightly. It can be explained that most of the red areas surrounded by green borders are correctly framed. Most of the boxes are correct but some symbols can be mistaken as texts. Under this rule, the recall rate drops so we need to improve the recall score by using the refined network.

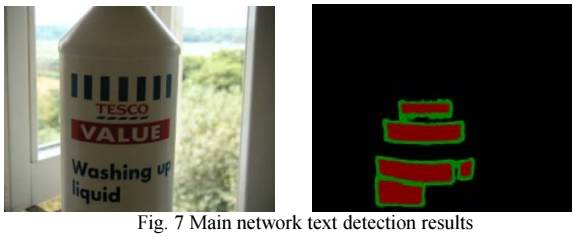

Fig. 7 Main network text detection results

TABLE I
THE SCORES OF MAIN NETWORK TEXT DETECTION

Method	Precision	Recall	F-Measure
Only the red area surrounded by the green area(A)	**90.63%**	81.59%	85.87%
All the red areas(B)	67.65%	**88.66%**	76.74%

B. *Refined network detection results and post-processing*

We show one result of refined network detection in Fig. 8. In fact, it is inferior to the main network in detecting the texts. Nevertheless, because of its region-based characteristics, it is not easy to have fragmentary predictions like the main network so the refined network makes up for the shortcomings of the main network. We compare the differences before and after the addition. In Fig. 8, the figure on the left is the result of the main network. The middle is the result of the refined network, and the right is the result of the addition of the refined network. We can see that the main network has some errors in the upper left corner of the figure, which will affect the performance if that part is not excluded. The refined network results are quite helpful in determining which region is wrong.

Because of the nature of semantic segmentation network, it is easy to produce small areas that are framed into small boxes. We remove them by using NMS. Table II shows the performance of various considered collocations. Also using the test set of ICDAR2013, we can find that the overall performance evaluated by F-Measure is progressing.

At the beginning, it has the highest precision and the lowest recall, and it is understandable that only the true parts are detected. However, it also decreases the number of detections. Adding the refined network increases the number of correct detections. Some wrong detection has also been added but the overall improvement can be observed. Finally, certain wrong detections are eliminated by NMS to achieve the best score. It should be noted that our model performs well in natural scenes, in which even dense and long Chinese text lines appear as shown in Fig. 9.

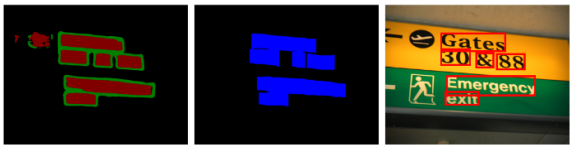

Fig. 8 An example of the refined network detection

Fig. 9 Some examples of detection in natural scenes

A comparison of our model with existing methods are demonstrated in Table III. Some of these models have a high recall rates because the precision can be increased by reducing the number of predictions. In other words, it is easy to make mistakes by making more predictions. Therefore, it is not easy to achieve high recall scores while maintaining high precision at the same time. The difference between our score and the best existing method is that we still frame certain non-text areas, such as artificial symbols. Other cases include wrongly merging some texts as they are too close. Currently we continue to improve the performance by including more suitable training data and a better refined region based network.

IV. CONCLUSION AND FUTURE WORK

In this paper, we proposed a method for detecting texts in images through a deep learning network architecture. The experimental results show that the proposed method performs very well. For complex natural images, most text blocks that can be recognized by the human eyes can be located. The strategy of combining the advantages of the two types of networks helps to achieve better results. There exists certain room for improvement. Possible directions include integrating the two models more closely to enhance the network performance and further enhancing the effectiveness of refined network.

ACKNOWLEDGMENT

This research is partially supported by the Ministry of Science and Technology under Grants MOST 109-2221-E-008-076, MOST 108-2634-F-008-008 and MOST 108-2221-E-003-027-MY3.

TABLE II
COMPARISON OF SCORES OF DIFFERENT COLLOCATIONS

Method	Precision	Recall	F-Measure
Main	**90.63%**	81.59%	85.87%
Main, refined	87.52%	86.54%	87.03%
Main,refined and NMS	89.06%	**87.14%**	**88.09%**

TABLE III
A COMPARISON OF OUR MODEL WITH OTHER MODELS

Method	Precision	Recall	F-Measure
CRAFT[7]	97.67%	92.40%	94.96%
FOTS[8]	94.63%	90.47%	92.50%
Mask Textspotter[9]	95.01%	88.27%	91.52%
RRPN-4[10]	94.91%	87.85%	91.25%
Ours	***89.06%***	***87.14%***	***88.09%***
TextBoxes_v2 [11]	91.92%	84.38%	87.67%
CRPN[12]	91.90%	83.08%	87.66%
CTPN[6]	92.77%	73.72%	82.15%

REFERENCES

[1] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie. 2017. Feature Pyramid Networks for Object Detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) , 936 - 944. DOI= 10.1109/CVPR.2017.106.

[2] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille. 2018. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence , 40(4), 834 - 848. DOI=10.1109/TPAMI.2017.2699184.

[3] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez, S. Robles, J. Mas, D. Fernandez, J. Almazan, L.P. de las Heras. 2013. ICDAR 2013 Robust Reading Competition. In Proc. 12th International Conference of Document Analysis and Recognition, IEEE CPS, pp. 1115-1124. DOI=10.1109/ICDAR.2013.221.

[4] K. He, X. Zhang, S. Ren, J. Sun. 2016. Deep residual learning for image recognition. DOI= 10.1109/CVPR.2016.90.

[5] G. J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla. 2008. Segmentation and Recognition Using Structure from Motion Point Clouds. Computer Vision – ECCV 2008 (Lecture Notes in Computer Science) , 5302, 44 - 57. DOI=10.1007/978-3-540-88682-2_5.

[6] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao. 2016 Detecting text in natural image with connectionist text proposal network. Computer Vision – ECCV 2016 (Lecture Notes in Computer Science) , 9912, 56 – 72. DOI=10.1007/978-3-319-46484-8_4.

[7] Y. Baek, B. Lee, D. Han, S. Yun, H. Lee. 2019. Character Region Awareness for Text Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 9365-9374).

[8] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, J. Yan. 2018 FOTS: Fast Oriented Text Spotting with a Unified Network. IEEE/CVF Conference on Computer Vision and Pattern Recognition , 5676 - 5685.DOI=10.1109/CVPR.2018.00595.

[9] P. Lyu, M. Liao, C. Yao, W. Wu, X. Bai. 2019. Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1. DOI= 10.1109/TPAMI.2019.2937086.

[10] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, X. Xue. 2018. Arbitrary-Oriented Scene Text Detection via Rotation Proposals. IEEE Transactions on Multimedia , 20(11), 3111 - 3122. DOI=10.1109/TMM.2018.2818020.

[11] M. Liao, B. Shi, X. Bai, X. Wang, W. Liu. 2017. TextBoxes: A Fast Text Detector with a Single Deep Neural Network. In Thirty-First AAAI Conference on Artificial Intelligence.
[12] L. Deng, Y. Gong, Y. Lin, J. Shuai, X. Tu, Y. Zhang, Z. Ma, M. Xie. 2019. Detecting Multi-Oriented Text with Corner-based Region Proposals. Neurocomputing , 334, 134 - 142. DOI=10.1016/j.neucom.2019.01.013.