Character Region Awareness for Text Detection – Supplementary Material –

Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee
Clova AI Research, NAVER Corp.
{youngmin.baek, bado.lee, dongyoon.han, sangdoo.yun, hwalsuk.lee}@navercorp.com

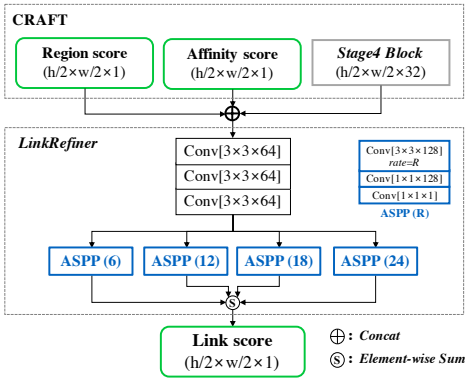

Figure 1. Schematic illustration of *LinkRefiner* architecture.

1. *LinkRefiner* for CTW-1500 dataset

CTW-1500 dataset [3] provides polygon-only annotations without text transcriptions. Furthermore, annotations of CTW-1500 are provided at the line-level and does not consider spaces as separation cues. This is far from our assumption of affinity, which is that the score for affinity is zero for characters with a space between them.

To obtain a single-long polygon from the detected characters, we employ a shallow network for link refinement, so called *LinkRefiner*. The architecture of the *LinkRefiner* is shown in Fig. 1. The input of the *LinkRefiner* is a concatenation of the *region score*, the *affinity score*, and the intermediate feature map from the network, that is the output of *Stage4* of the original CRAFT model. Atrous Spatial Pyramid Pooling (ASPP) in [1] is adopted to ensure a large receptive field for combining distant characters and words onto the same text line.

For the ground truth of the *LinkRefiner*, lines are simply drawn between the centers of the paired control points of the annotated polygons, which is similar to the text line generation used in [2]. The width of each line is proportional to the distance between paired control points. The ground truth generation for the *LinkRefiner* is illustrated in Fig. 2. The output of the model is called the *link score*. For training, only the *LinkRefiner* is trained on the CTW-1500 training dataset, while freezing CRAFT.

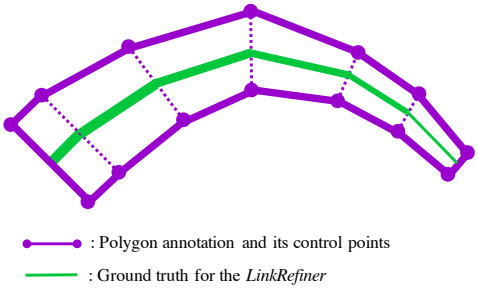

Figure 2. Ground truth generation for *LinkRefiner*.

After training, we have the outputs produced by the model, which are the *region score*, the *affinity score*, and the *link score*. Here, the *link score* is used instead of the original *affinity score*, and the text polygon is obtained entirely through the same process as done with TotalText. The CRAFT model localizes the individual characters, and the *LinkRefiner* model combines the characters as well as the words separated by spaces, which are required by the CTW-1500 evaluation.

The results on the CTW-1500 dataset are shown in Fig. 3. Very challenging image samples with long and curved texts are successfully detected by the proposed method. Moreover, with our polygon representation, the curved images can be rectified into straight text images, which are also shown in Fig. 3. We believe this ability for rectification can further be of use for recognition tasks.

References

[1] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *PAMI*, 40(4):834–848, 2018. 1

[2] T. He, W. Huang, Y. Qiao, and J. Yao. Accurate text localization in natural image with cascaded convolutional text network. *arXiv preprint arXiv:1603.09423*, 2016. 1

[3] L. Yuliang, J. Lianwen, Z. Shuaitao, and Z. Sheng. Detecting curve text in the wild: New dataset and new solution. *arXiv preprint arXiv:1712.02170*, 2017. 1

Figure 3. Results on CTW-1500 dataset. For each cluster: the input image (top), region score (middle left), link score (middle right), and the resulting rectified polygons for curved texts (bottom, below arrow) are shown. Note that the affinity scores are not rendered and are unused in the CTW-1500 dataset.