EAFormer: Scene Text Segmentation with Edge-Aware Transformers –Supplementary Material–

Haiyang Yu, Teng Fu, Bin Li*, and Xiangyang Xue

Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University
{hyyu20, libin, xyxue}@fudan.edu.cn, tfu23@m.fudan.edu.cn

1 Details about the backbone of text detection.

In text edge extractor, we employ a ResNet-based backbone for text detection. Similar to ResNet-18, each layer contains two residual blocks. However, there are two differences between ResNet-18 and the adopted backbone: 1) The stride of the first convolutional layer before the four residual layers is set to 4 rather than 2, which results in the feature maps with a smaller size and saves time cost in text detection. 2) To build a more lightweight backbone, we set the channels of four layers to {32, 64, 160, 256}. Since the coarse text detection results are sufficient for filtering out the edges of non-text areas, it is unnecessary to introduce more learnable parameters in the text edge extractor module.

We also conduct experiments to investigate the influence when using the more complex backbones for text detection. Through the results shown in Table 1, we observe that the backbone with more parameters may not be beneficial for the text segmentation performance. One possible reason for these results is that a more complex backbone makes the model focus more on optimizing the text detection module, thus interfering the performance of text segmentation.

Backbone	TextSeg		BTS	
	fgIoU	F-score	fgIoU	F-score
ResNet-18	87.60	0.934	86.32	0.901
ResNet-34	87.15	0.928	87.03	0.915
ResNet-50	87.26	0.930	87.51	0.928
Ours	**88.06**	**0.939**	**88.08**	**0.937**

Table 1: The experimental results of choosing backbone. We observe that the proposed lightweight backbone for text detection is more efficient.

* Corresponding Author

2 Spatial reduction in self-attention.

In [6], the authors introduce the spatial reduction strategy to reduce the computation cost of self-attention. In this paper, we follow it to perform the spatial reduction operation on Key and Value:

$$\mathrm{SR}(x) = \mathrm{Norm}(\mathrm{Reshape}(\mathbf{I}, Ratio)W_r), \tag{1}$$

where $\mathbf{I}$ and $Ratio$ represent the input feature maps and the reduction ratio, respectively. Reshape(x, r) is used to reshape the features x from $(HW) \times C$ to $\frac{HW}{r^2} \times (r^2 C)$, and W_r is the parameters of linear for further reducing the channel dimension of the input to C. Norm$(\cdot)$ denotes the layer normalization [1].

3 Examples of adopted datasets.

In this paper, we have conducted experiments on six publicly available datasets: ICDAR13 FST [5], COCO_TS [2], MLT_S [3], Total-Text [4], TextSeg [8], and BTS [9]. Some examples of each datasets are shown in Figure 1. As discussed in the main text, the annotations of COCO_TS and MLT_S are inaccurate since they are obtained through a weakly-supervised method. Thus, we have re-annotated these two datasets for more convincing experimental results.

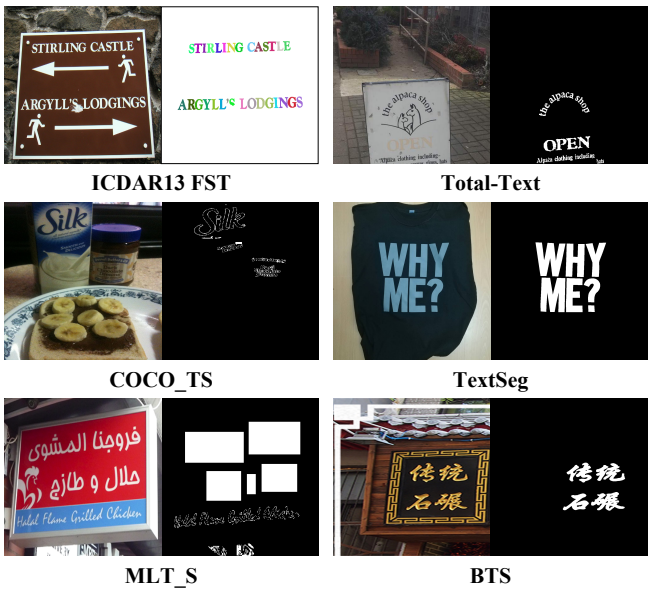

Fig. 1: Some examples of the adopted datasets. Considering that the annotations of COCO_TS and MLT_S are inaccurate, we have re-annotated all samples of them.

4 More discussions about edge filtering.

In the proposed EAFormer, we filter out irrelevant edge areas because we find that the Canny algorithm can extract a lot of edge information of non-text areas, as shown in the middle column of Figure 2. If we do not filter out irrelevant edge information, the model will be misled by the input edge information, that is, it believes that the areas with edge information have a high probability of belonging to text areas, as shown in the third column of Figure 2.

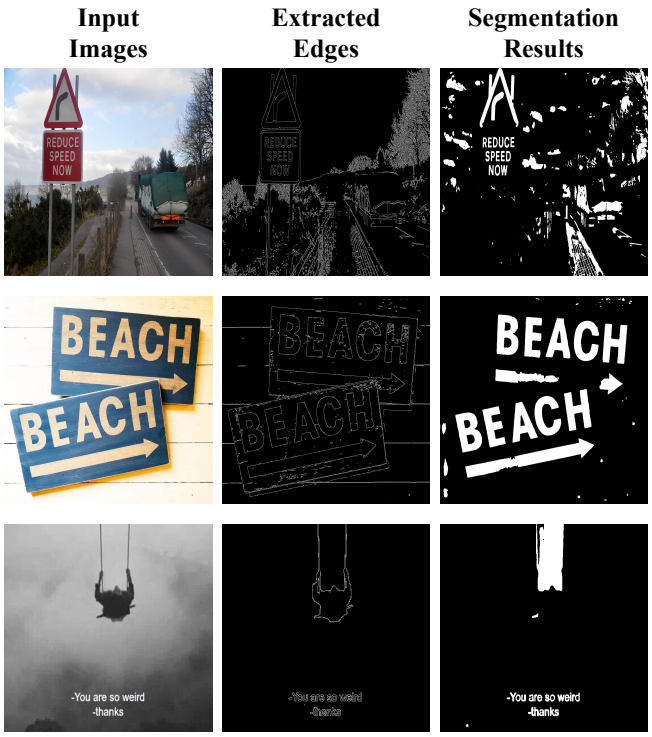

Fig. 2: The result visualization when all detected edges are used in EAFormer.

5 Introducing text-edge information at other stages.

As discussed in the main texts, we choose to introduce text-edge information at the first stage of the transformer-based encoder since the first stage focuses more on the edge information. We also try to introduce the text-edge information at other stages and conduct related experiments. The results shown in Table 2 demonstrate that it is most beneficial to introduce text edge information at the first stage. Specifically, compared with the baseline model on TextSeg, the proposed EAFormer can achieve an improvement of 3.47%/2.3% in fgIoU/F-score. In addition, we observe that when introducing the edge information at the third and fourth stages, the proposed EAFormer can hardly achieve better performance than the baseline model.

i-th Stage	TextSeg		BTS	
	fgIoU	F-score	fgIoU	F-score
Baseline	84.59	0.916	84.99	0.908
1	88.06	0.939	88.08	0.937
2	86.62	0.921	85.82	0.901
3	85.01	0.918	85.25	0.898
4	83.28	0.910	83.10	0.871

Table 2: The experimental results of introducing edge information at different stages.

6 More visualizations of experimental results.

The proposed EAFormer aims to make the model focus more on the text-edge areas, thus achieving more accurate segmentation results at the text edges. To verify the effectiveness of EAFormer, we visualize more segmentation results. As shown in Figure 3-8, the proposed EAFormer can indeed obtain better results compared with the previous SOTA method TextFormer [7]. Specifically, on Total-Text, the results of the proposed EAFormer can segment texts more accurately than the previous SOTA method. Meanwhile, benefiting from the proposed text-edge extractor, some texts with small scales can be perceived robustly.

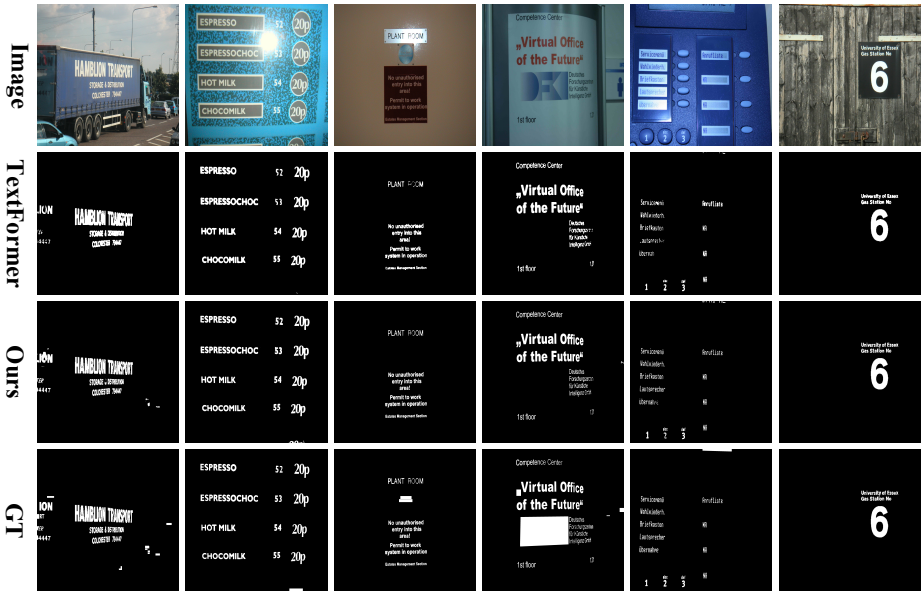

Fig. 3: Segmentation results of ICDAR13 FST.

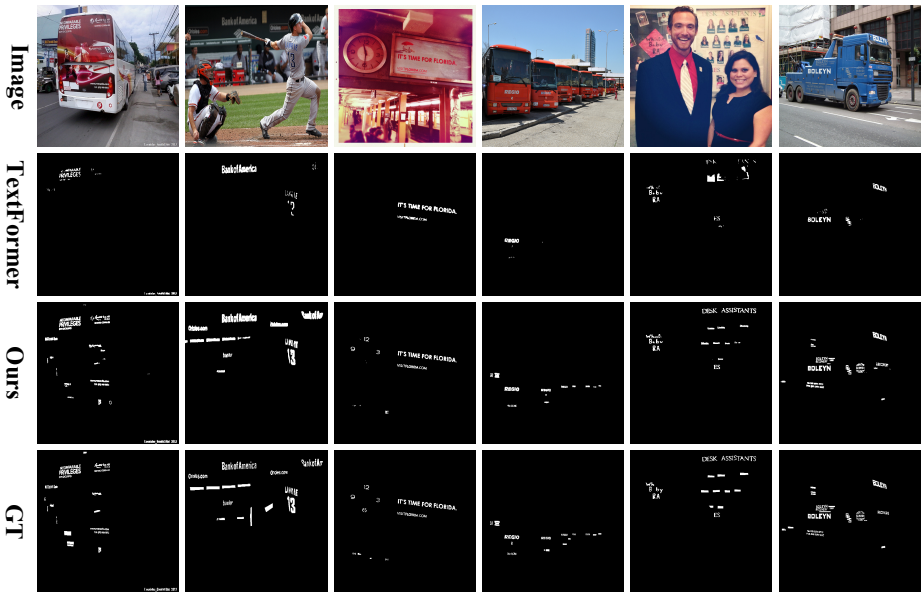

Fig. 4: Segmentation results of COCO_TS.

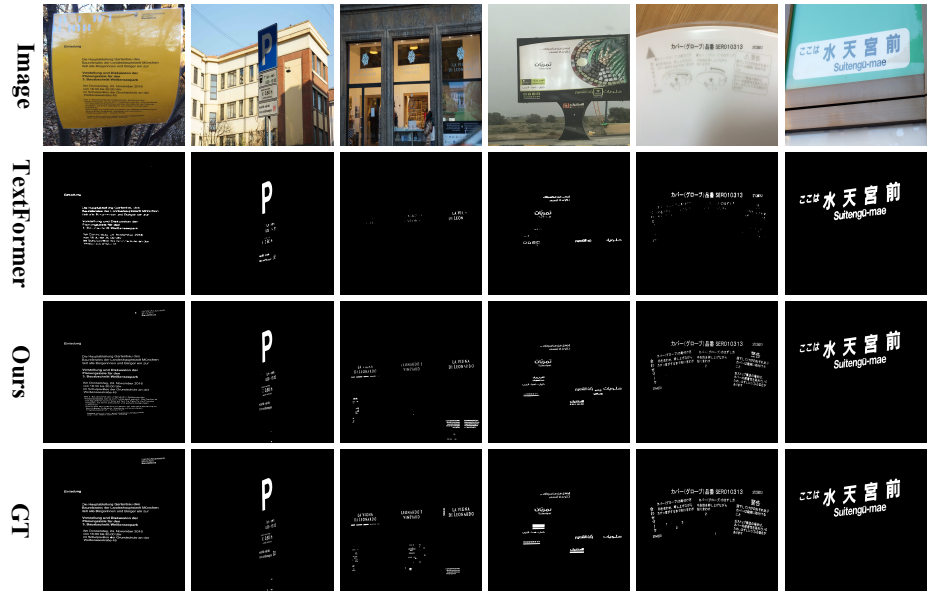

Fig. 5: Segmentation results of MLT_S.

Fig. 6: Segmentation results of Total-Text.

Fig. 7: Segmentation results of TextSeg.

Fig. 8: Segmentation results of BTS.

References

1. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016) 2
2. Bonechi, S., Andreini, P., Bianchini, M., Scarselli, F.: Coco_ts dataset: pixel–level annotations based on weak supervision for scene text segmentation. In: International Conference on Artificial Neural Networks. pp. 238–250. Springer (2019) 2
3. Bonechi, S., Bianchini, M., Scarselli, F., Andreini, P.: Weak supervision for generating pixel–level annotations in scene text segmentation. Pattern Recognition Letters **138**, 1–7 (2020) 2
4. Ch'ng, C.K., Chan, C.S.: Total-text: A comprehensive dataset for scene text detection and recognition. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR). vol. 1, pp. 935–942. IEEE (2017) 2
5. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: Icdar 2013 robust reading competition. In: 2013 12th international conference on document analysis and recognition. pp. 1484–1493. IEEE (2013) 2
6. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: ICCV. pp. 568–578 (2021) 2
7. Wang, X., Wu, C., Yu, H., Li, B., Xue, X.: Textformer: Component-aware text segmentation with transformer. In: 2023 IEEE International Conference on Multimedia and Expo (ICME). pp. 1877–1882. IEEE (2023) 4
8. Xu, X., Zhang, Z., Wang, Z., Price, B., Wang, Z., Shi, H.: Rethinking text segmentation: A novel dataset and a text-specific refinement approach. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12045–12055 (2021) 2
9. Xu, X., Qi, Z., Ma, J., Zhang, H., Shan, Y., Qie, X.: Bts: a bi-lingual benchmark for text segmentation in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19152–19162 (2022) 2