Appendix for "Boundary-enhanced Co-training for Weakly Supervised Semantic Segmentation"

Contents

A. Experiments

A.1. More Implementation Details

Visualization of Training samples. Figure 1 shows some training samples of *BECO*. Note that the *BECO* model is jointly fed the boundary-unknown (*i.e.*, original) images and the boundary-aware images. For boundary-unknown images, their pseudo-labels and corresponding confidence masks are generated offline from the first stage of WSSS. And the boundary map of them is an all-zero matrix. For boundary-aware images, their pseudo-labels are generated online from the ensemble of predictions from two networks. From (c) confidence masks, we can observe that the uncertain pixels are concentrated on boundary areas and pixels with high confidence tend to be correct and inside the object. And the boundary map only represents the boundary pixels of the copy-pasted class mask.

Illustration of Generating Boundary Map. Here, we provide the implementation illustration for generating the boundary map. We apply dilation and erosion on the given class mask $\overline{\boldsymbol{M}}_{ch1}$ to obtain the dilated variant $\overline{\boldsymbol{M}}_{chd1}$ and eroded variant $\overline{\boldsymbol{M}}_{che1}$, respectively, where the kernel size of dilation and erosion is set as 3. As shown in Figure 2, we then perform a subtraction operation between $\overline{\boldsymbol{M}}_{chd1}$ and $\overline{\boldsymbol{M}}_{chde1}$ to obtain the boundary map $\boldsymbol{B}^{'}$.

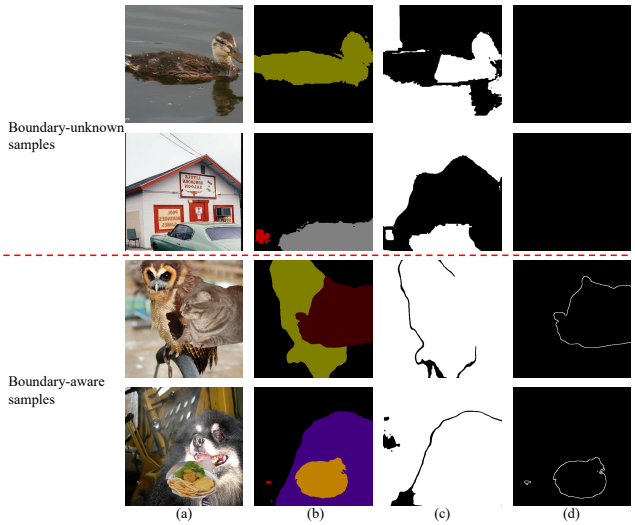

Figure 1. Visualization of training samples. (a) Input images $\boldsymbol{X}$, (b) Pseudo-labels $\boldsymbol{Y}$, (c) Confidence masks $\boldsymbol{M}$, (d) Boundary maps $\boldsymbol{B}$. The boundary map of the boundary-unknown sample is an all-zero matrix. Best viewed in color.

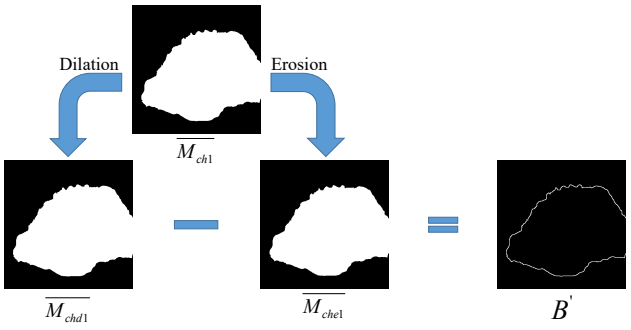

Figure 2. Illustration of the boundary map generation. Best viewed in color.

A.2. Additional Results Using Different Deeplab

Some previous WSSS works adopt the standard DeeplabV2 as the segmentation network, which uses ResNet101 with an output stride of 8. Here, we also provide additional results of our method using DeeplabV2 framework. As shown in Table 1, the *BECO* using DeeplabV2 and DeeplabV3+ achieve similar performance on PAS-

CAL VOC 2012 *val* set, (72.3% and 72.1% mIoU, respectively), outperforming the state-of-the-art methods. And our proposed COT and *BECO** surpass the ENSEMBLE by 1.8% and 4.5% on average, respectively. Considering that DeeplabV2 with an output stride of 8 requires more GPU memory and has a slower training/inference speed, we use DeeplabV3+ with an output stride of 16 as the default framework.

Table 1. Effectiveness using different Deeplab in terms of mIoU(%) on VOC 2012 *val* set. All Deeplab versions use the same backbone, *i.e.*, ResNet101. *BECO**: *BECO* without label refinement.

Method	DeeplabV2	DeeplabV3+
Baseline	65.6	65.1
ENSEMBLE	65.9 (+0.3)	66.2 (+1.1)
COT	67.5(+1.6)	68.2 (+2.0)
*BECO**	70.2(+2.7)	70.9 (+2.7)
BECO	72.3(+2.1)	72.1 (+1.2)

A.3. Boundary Weight λ for the *BECO* Loss

We report the ablation result for λ of *BECO* on PASCAL VOC dataset in Figure 3. We observe that our model is robust to the choice of λ.

Table 2. Ablation study for r in terms of mIou(%) on PASCAL VOC 2012. We fixed the other hyperparameters to the default setting.

r	0%	40%	50%	60%	100%
mIoU	NaN	70.5	70.9	70.1	67.6

A.4. Ratio r for Offline Confidence Mask Generation

The offline confidence mask regards the pixels with the top r confidence in the same category as high confidence and the rest are as low confidence. According to the proposed co-training paradigm, only the offline pseudo-labels of high-confidence pixels participate in training, while the remaining pixels are supervised by the online prediction of another network. Here, we report the ablation for r of the offline confidence mask generation in Table 2. A smaller r indicates that there are fewer offline pseudo-label pixels involved in the *BECO* training, which loses too much information from the first stage of WSSS and leads to the underfitting of the model. When r becomes 0%, the model performs self-learning without using offline pseudo-labels, which leads to model non-convergence. A larger r indicates that more offline pseudo-label pixels are involved in training, which leads to *BECO* overfitting to the increased noisy annotations. As a result, we empirically choose r=50% for all experiments on PASCAL VOC 2012 dataset and MS COCO 2014 dataset.

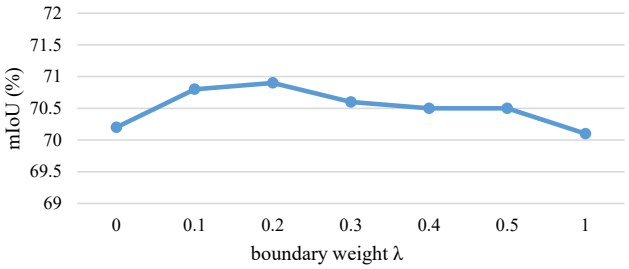

Figure 3. Ablation study for λ on PASCAL VOC 2012. We fixed the other hyperparameters to the default settings.

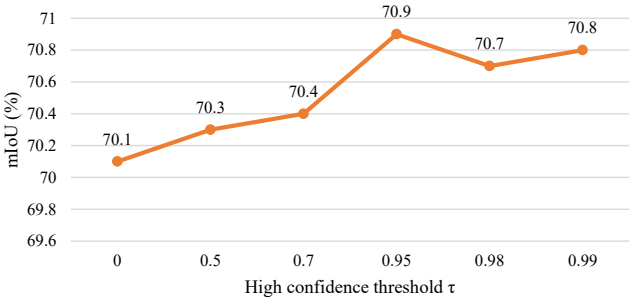

Figure 4. Ablation study for τ on PASCAL VOC 2012. We fixed the other hyperparameters to the default settings.

A.5. Threshold τ for Generating Online Confidence Mask

We report the ablation result for τ of online confidence mask generation in Figure 4. Generating confidence masks by large τ values is beneficial to accurate boundary construction. We can observe that results are robust to the choice of large τ value. If the value of τ is too small (*e.g.*, 0, 0.5), some noise that is not the real boundary will be introduced, resulting in a degraded boundary quality. When the τ value is 0, our boundary construction strategy is similar to Classmix. However, the naive copy-and-paste techniques cannot guarantee to construct the boundary of the actual objects. Compared with $\tau = 0$, constructing boundaries by a large threshold $\tau = 0.95$ is beneficial for the network to learn the edge regions with more accurate annotations and increases by 0.8% mIoU.

A.6. Analysis of the Coupling Effect of Co-training

In the proposed co-training paradigm, the same input is fed into the siamese networks which interact with each other by the generated pseudo-supervisions. One concern may arise that the siamese networks with the same architecture are easy to make the same predictions suffering from a trivial solution, that is the two networks become coupled and result in confirmation bias. To analyze the coupling effect of two networks in COT, we train two models where the siamese networks with different initialization and the same initialization are denoted as COT-D and COT-S, re-

Table 3. Quantitive results of boundary in terms of maximal F-measure(%) on SBD *val* set.

Method	aeroplane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motorbike	person	plant	sheep	sofa	train	tv	mean
Baseline	**81.3**	68	82.3	63	54.8	85.7	73	88.6	54.8	**88.9**	55.9	87.1	83.2	77.7	71.4	58.3	82.8	70.7	**76.8**	65.4	73.49
BECO	80.5	**70.9**	**83.1**	**65.6**	**57.1**	**86.4**	**76.1**	**92.8**	**55.4**	86.8	**60.2**	**91.1**	**84.2**	**79.2**	**73.6**	**64.9**	**84.7**	**74.4**	68.9	**66.4**	**75.12**

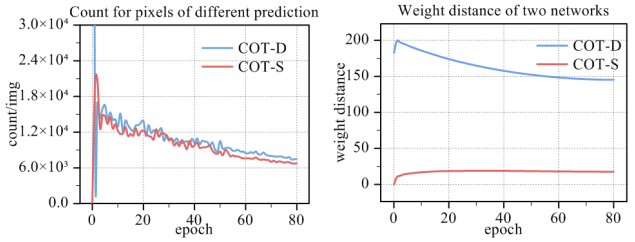

Figure 5. The co-training networks using different initialization (COT-D) vs. the same initialization (COT-S). Left: the average count of different predictions per image of the two networks in COT-D and COT-S keep large values. Right: The weight distance of the two networks in COT-D and COT-S both do not converge to zero, while the weights of the two networks in COT-D keep a larger distance than those of COT-S.

spectively. COT-D and COT-S achieve similar performance on PASCAL VOC 2012 *val* set(68.2% and 67.9% mIoU, respectively). We then calculate the euclidean distance of the weights and the average count of different predictions per image (*i.e.*, the prediction distance) between the two networks in each model. As shown in Figure 5, the prediction distances of COT-D and COT-S still maintain large values even in the last training epochs. Although the two networks are initialized identically, the weight distance of COT-S still does not converge to zero. And the weight distance of COT-D is larger than that of COT-S. These results demonstrate that the two independent networks in the proposed co-training paradigm are loosely coupled. We argue that different initialization and dropout layers in the networks resemble different network perturbations, which alleviates the coupling effect between the two networks. Due to the slightly superior performance of COT-D, we initialize two independent networks differently by default in all experiments.

A.7. Quantitative Results of Boundary Improvement.

To provide quantitative results of boundary improvement, we also evaluate *BECO*'s boundary quality on SBD benchmark [1], which contains semantic boundary annotations of 11355 images from the PASCAL VOC 2011 dataset. We test the predicted boundary maps of baseline and *BECO* with maximal F-measure on SBD *val* set containing 2857 images. Table 3 shows that *BECO* performs substantially better than the baseline on most of the semantic categories and on average.

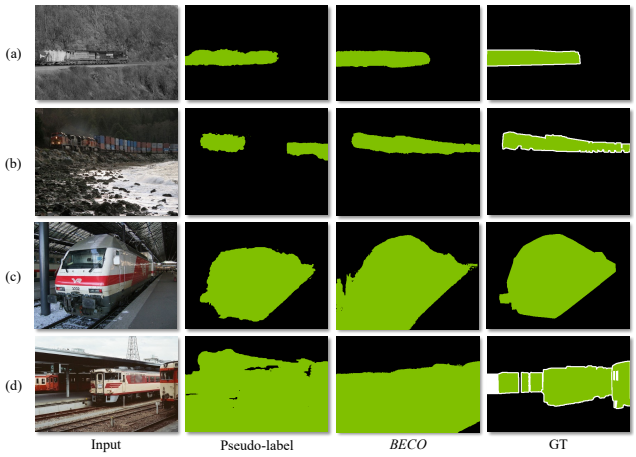

Figure 6. Visualization of the segmentation results of *BECO*. Due to the co-occurring pixels from non-target objects(*e.g.*, a railroad), *BECO* would fail to assign the co-occurring pixels to the background class in some cases, *e.g.*, (c) and (d).

A.8. Visualization on MS COCO 2014

We provide additional segmentation results of *BECO* for several examples on the MS COCO 2014 *val* set, as shown in Figure 7.

B. Limitations

Compared to the traditional training paradigm (training a segmentation network with all pseudo-labels), *BECO* aims to exploit the generalization ability of the network to learn low-confidence parts, while the high-confidence parts are supervised by pseudo-labels. In Section 4.2 of the main paper, we have demonstrated the effectiveness of *BECO*. However, with the improvement of *BECO*'s learning ability, in some exceptional cases, *BECO* would also learn some "system errors". For example, as shown in Figure 6 (c) and (d), *BECO* is more inclined to identify "railroad" as "train". We argue that it is difficult to distinguish co-occurring pixels of non-target objects from a target object without additional prior knowledge due to the task setting and dataset limitations of WSSS.

References

[1] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 3

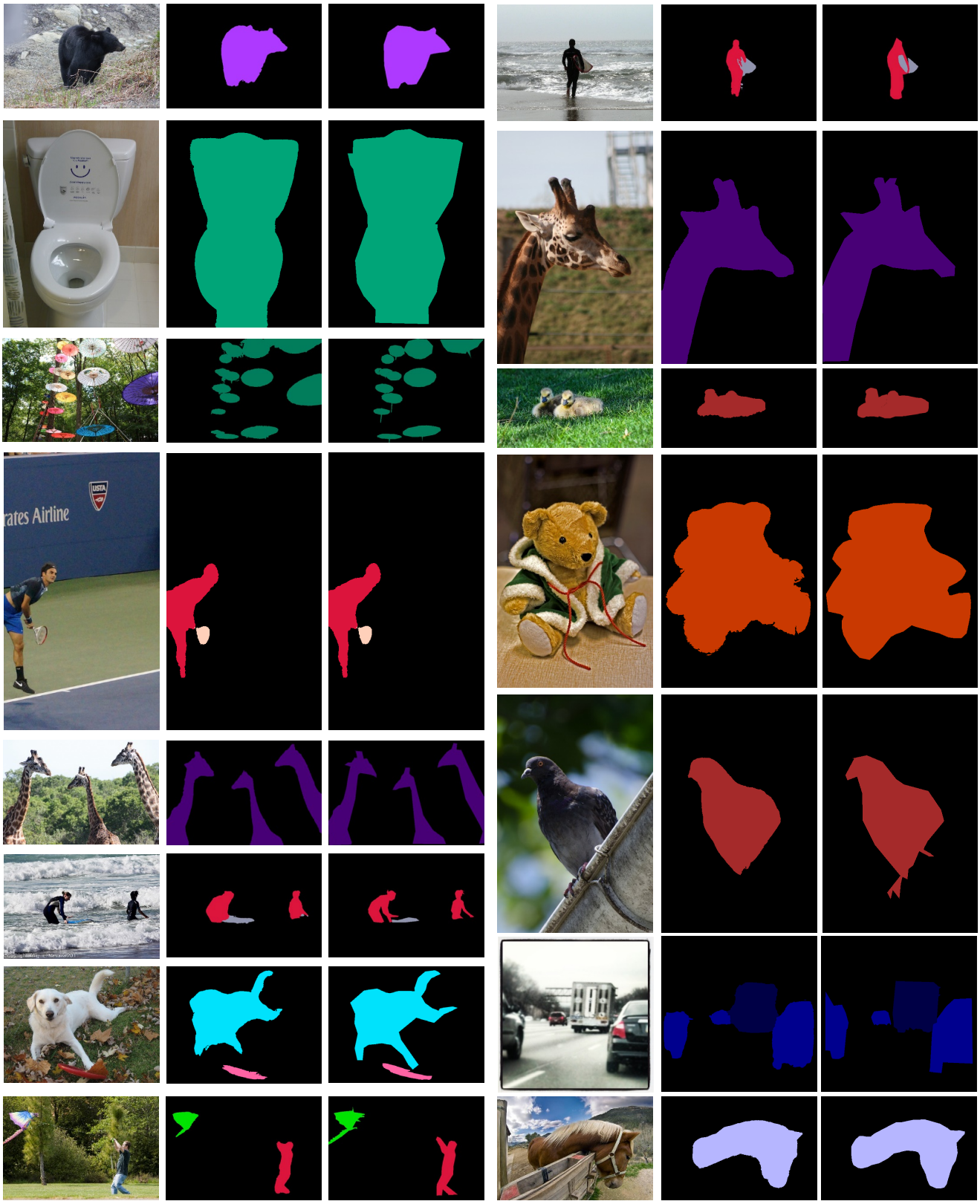

Figure 7. Qualitative segmentation results on the MS COCO 2014 *val* set.