Supplementary of Weakly Supervised Semantic Segmentation via Progressive Confidence Region Expansion

Xiangfeng Xu[1*] Pinyi Zhang[1*] Wenxuan Huang[1] Yunhang Shen[2] Haosheng Chen[1]
Jingzhong Lin[1] Wei Li[3] Gaoqi He[1] Jiao Xie[1] Shaohui Lin[1,4✉]
[1]East China Normal University [2]Xiamen University [3]Huawei Noah's Ark Lab
[4]Key Laboratory of Advanced Theory and Application in Statistics and Data ScienceMinistry of Education, China

1. More Training Details.

The segmentation head is composed of four simple 3×3 convolutional layers. Input images are processed through a random augmentation strategy involving color jitter, random scaling, and random flip. For the experiments conducted on the VOC 2012 dataset, we set the batch size to 4. The models are trained for $20,000$ iterations, with a warm-up phase of $2,000$ iterations for the segmentation heads. For the COCO dataset, the batch size is set to 8 and the models are trained for $80,000$ iterations with $8,000$ iterations warmed up for the segmentation head. The radius r, as defined in Eq. (5), is initialized to 1 and progressively expanded to cover the entire target region. The expansion follows a cosine updating strategy applied between 10% and 40% of the total training iterations. Following previous works [1–3], we also incorporate the Patch Token Contrast (PTC) loss [1] to ensure a fair comparison with existing methods.

2. Effect of Hyper-parameters

In this section, we conduct experiments on the effects of other hyper-parameters, including the binary mask threshold τ_{bin} in Eq. 7, threshold β to obtain pseudo labels, max iteration number T, the smoothing factor λ_p in Eq. 12 and the loss weighting parameters $\lambda_1, \lambda_2, \lambda_3$. All experiments were conducted on the PASCAL VOC 2012 validation set without applying CRF post-processing.

Effect on the binary mask threshold τ_{bin}. Tab. 1a explores the impact of the binarization threshold τ_{bin}, which is used to convert masks from CRME into binary masks for ROR. Experimental results show that $\tau_{\text{bin}} = 0.7$ achieves the optimal balance for this process.

Effect on the Threshold β. Tab. 1b reports the effect of the threshold β, which determines the boundary between the foreground and background regions in CAM-based pseudo-labels. The results show that setting β to 0.5 provides the most reliable balance for pseudo-label generation.

Effect on the max iteration number T in Progressive Mask Expansion and Combination. Tab. 1c summarizes the different max iteration numbers for progressive mask expansion in each class. Setting $T = 5$ achieves the best performance by progressively refining and expanding high-confidence regions for each class, which ensures comprehensive coverage of multi-instance objects within a single class. Smaller T (*e.g.*, 3) might miss less prominent instances, while larger iteration numbers (*e.g.*, 10) will lead to redundant expansions, increasing the risk of background interference or overlap between instances.

Effect on the Smoothing factor λ_p. Tab. 1d presents the results of different smoothing factors λ_p, which are used to update class prototypes to balance between the historical prototypes and the current class tokens. The setting of $\lambda_p = 0.99$ achieves the best performance, compared to other numbers.

Effect on the Loss weighting parameters. Tab. 1e, 1f and 1g report the impact of the loss weighting parameters λ_1, λ_2, and λ_3, which balance the contributions of the confidence loss $\mathcal{L}_{\text{conf}}$, refined mask loss $\mathcal{L}_{\text{refine}}$, and prototype alignment loss $\mathcal{L}_{\text{CPE}}$, respectively. The optimal settings are $\lambda_1 = 2.0$, $\lambda_2 = 1.0$, and $\lambda_3 = 0.02$, ensuring an effective balance across all loss components for robust training.

Effect on the radius r. In Tab. 1h, fixed $r = 5$ keeps the radius constant, w/o r disables radius constraints, and Cosine r dynamically updates the radius using a cosine schedule.Experimental results show that Cosine r r achieves the best performance by progressively expanding the region coverage while maintaining precision.

3. Additional Quantitative and Qualitative Results

Per-Class Segmentation Comparison. Tab. 2 illustrates that our method achieves leading performance in 12 out of 21 classes, outperforming previous SOTA WSSS ap-

*Equal contribution. ✉Corresponding author.

proaches. Specifically, our method achieves the highest performance with 67.8% to segment the sofa, which outperforms the best SOTA DuPL by 4.4%. In addition, we achieve the highest mIoU of 75.5%, compared to all SOTA methods. The improvement reflects the effectiveness of our Progressive Confidence Region Expansion framework in generating precise segmentation masks, which mitigates the over-expansion issue.

Per-Class OA Rate Comparison. Fig. 1 compares the Over-Activation (OA) rates for each class between our method and DuPL [2] on the PASCAL VOC 2012 validation set. We observe that our method consistently reduces the OA rate across most classes. These results highlight the effectiveness of our method in refining segmentation masks and preventing the unintended spread of high activation values to background regions.

CAM Visualization Comparison on MS COCO. Fig. 2 compares the Class Activation Maps (CAMs) generated by ToCo [1], DuPL [2], and our method on the MS COCO dataset. The visualizations demonstrate that our method produces more accurate and focused CAMs, effectively localizing target objects while reducing over-expansion into irrelevant background regions.

More Segmentation Visualization Results. Fig. 3 and 4 showcase segmentation results on PASCAL VOC 2012 and MS COCO, respectively. Our method achieves more accurate and complete segmentation compared to ToCo [1] and DuPL [2].

References

[1] Lixiang Ru, Heliang Zheng, Yibing Zhan, and Bo Du. Token contrast for weakly-supervised semantic segmentation. In *CVPR*, pages 3093–3102, 2023.

[2] Yuanchen Wu, Xichen Ye, Kequan Yang, Jide Li, and Xiaoqiang Li. Dupl: Dual student with trustworthy progressive learning for robust weakly supervised semantic segmentation. In *CVPR*, pages 3534–3543, 2024.

[3] Lixiang Ru, Yibing Zhan, Baosheng Yu, and Bo Du. Learning affinity from attention: End-to-end weakly-supervised semantic segmentation with transformers. In *CVPR*, pages 16846–16855, 2022.

[4] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *CVPR*, pages 4253–4262, 2020.

τ_{bin}	CAM	Seg.
0.8	75.8	73.4
0.7	76.3	73.8
0.6	75.7	73.6
0.5	75.3	73.1

(a) **Binarization thresholds**.

β	CAM	Seg.
0.60	74.3	72.9
0.55	75.1	73.6
0.5	76.3	73.8
0.45	75.2	73.4

(b) **Background thresholds**.

T	CAM	Seg.
10	75.4	72.8
5	76.3	73.8
3	75.8	73.8
1	75.0	73.4

(c) **Iterations of PMEC**.

λ_p	CAM	Seg.
0.999	75.0	72.6
0.99	76.3	73.8
0.9	75.2	73.2
0.5	75.3	73.3

(d) **Smoothing factor in CPE**.

λ_1	CAM	Seg.
2.5	75.9	73.6
1.5	75.3	73.3
1.0	76.3	73.8
0.5	75.7	73.2

(e) **Loss weight of $\mathcal{L}_{\text{conf}}$**.

λ_2	CAM	Seg.
3.0	76.1	73.4
2.0	76.3	73.8
1.0	75.6	73.1
0.5	75.7	73.4

(f) **Loss weight of $\mathcal{L}_{\text{refine}}$**.

λ_3	CAM	Seg.
1.0	74.8	71.8
0.1	75.0	73.1
0.02	76.3	73.8
0.005	74.5	73.0

(g) **Loss weight of $\mathcal{L}_{\text{CPE}}$**.

r setting	CAM	Seg.
w/o r	75.0	72.6
Fixed $r = 5$	75.3	73.1
Cosine r	76.3	73.8

(h) **Radius r setting.**

Table 1. **Ablation study of hyper-parameters.**

	bkg	*aero*	*bicycle*	*bird*	*boat*	*bottle*	*bus*	*car*	*cat*	*chair*	*cow*	*table*	*dog*	*horse*	*motor*	*person*	*plant*	*sheep*	*sofa*	*train*	*tv*	*mIoU*
1Stage [4]	88.7	70.4	35.1	75.7	51.9	65.8	71.9	64.2	81.1	30.8	73.3	28.1	81.6	69.1	62.6	74.8	48.6	71.0	40.1	68.5	64.3	62.7
AFA [3]	89.9	79.5	31.2	80.7	67.2	61.9	81.4	65.4	82.3	28.7	83.4	41.6	82.2	75.9	70.2	69.4	53.0	85.9	44.1	64.2	50.9	66.0
ToCo [1]	91.1	80.6	**48.7**	68.6	45.4	79.6	87.4	**83.3**	89.9	35.8	84.7	60.5	83.7	83.2	76.8	83.0	56.6	87.9	43.5	60.5	63.1	71.1
DuPL [2]	91.8	77.8	47.1	81.7	58.9	78.6	**88.8**	77.6	91.9	38.2	**91.5**	55.5	**88.0**	**90.0**	77.7	**85.9**	60.7	**92.7**	54.0	**66.1**	45.5	73.3
Ours	**92.8**	**84.4**	41.1	**83.3**	**67.8**	**79.7**	88.7	82.4	**91.9**	**42.5**	88.0	**64.8**	87.6	88.0	**79.0**	83.3	**65.1**	90.5	**58.4**	61.7	**64.8**	**75.5**

Table 2. Performance on per-class segmentation on VOC validation set.

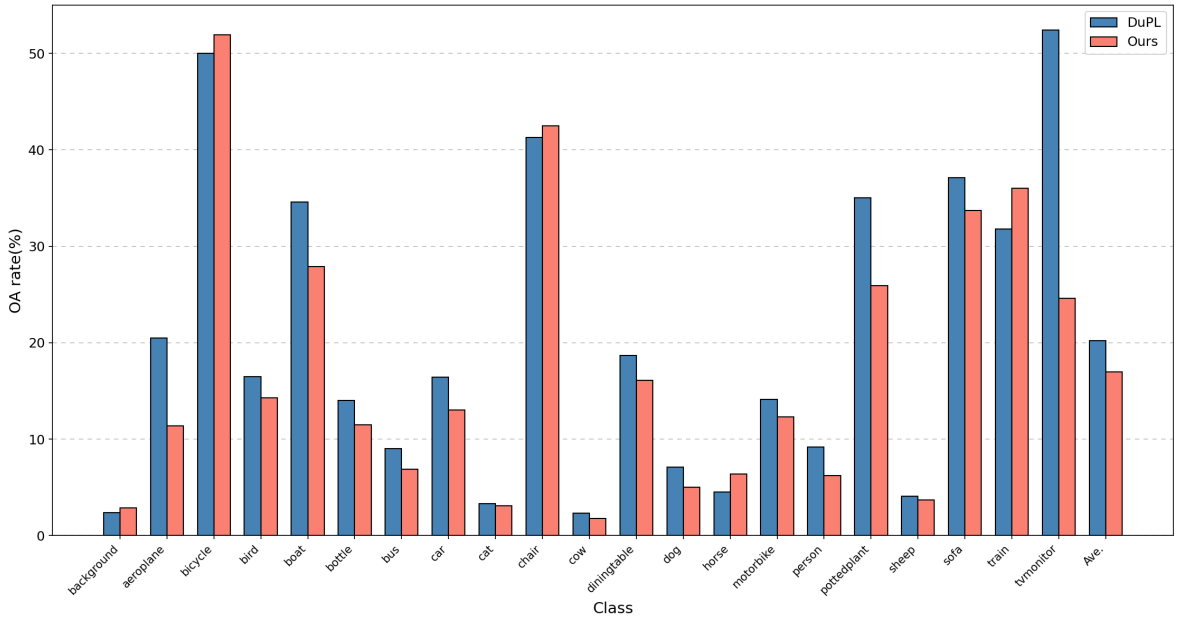

Figure 1. Per-Class OA Rate Comparison.

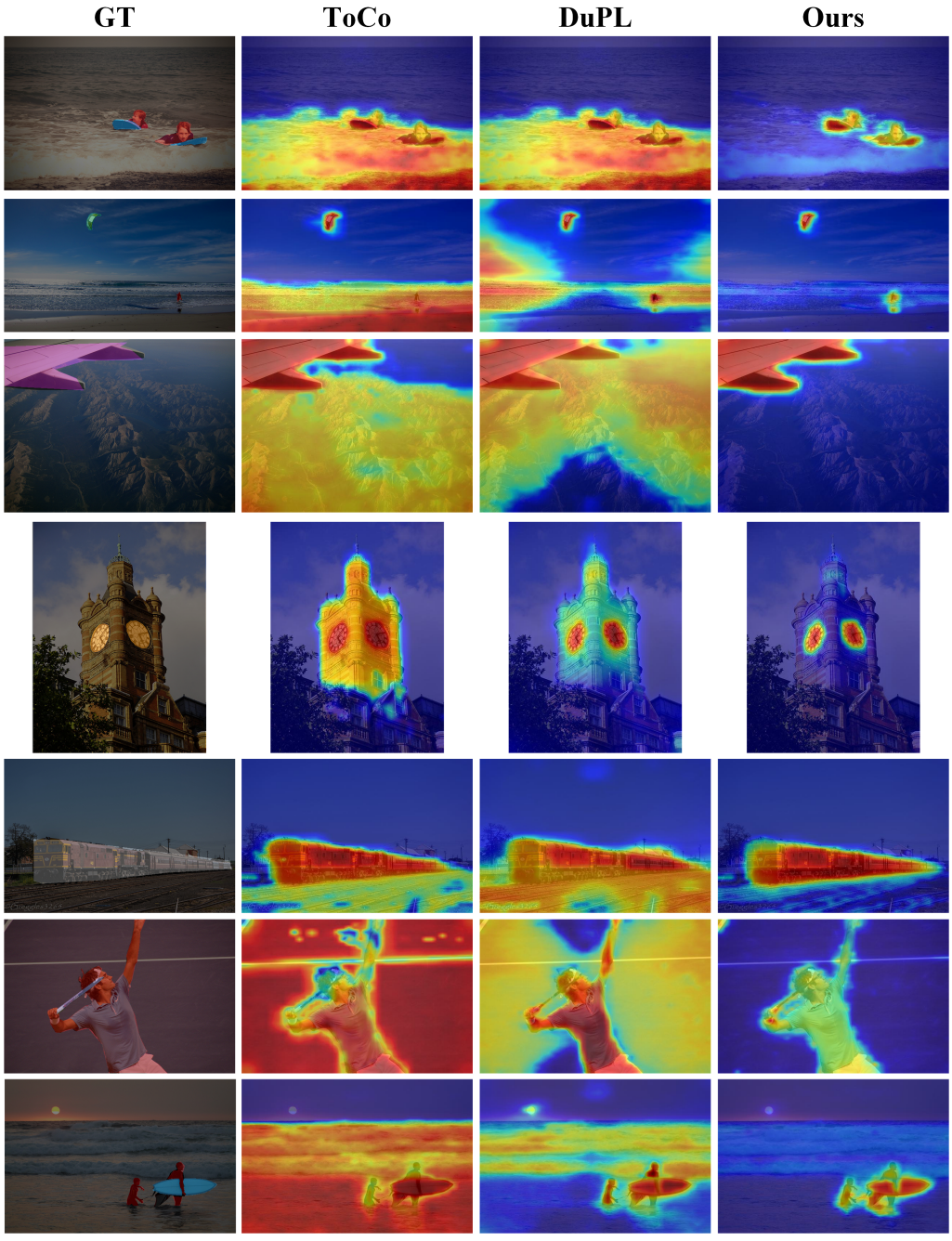

Figure 2. More visualization results of CAM on MS COCO.

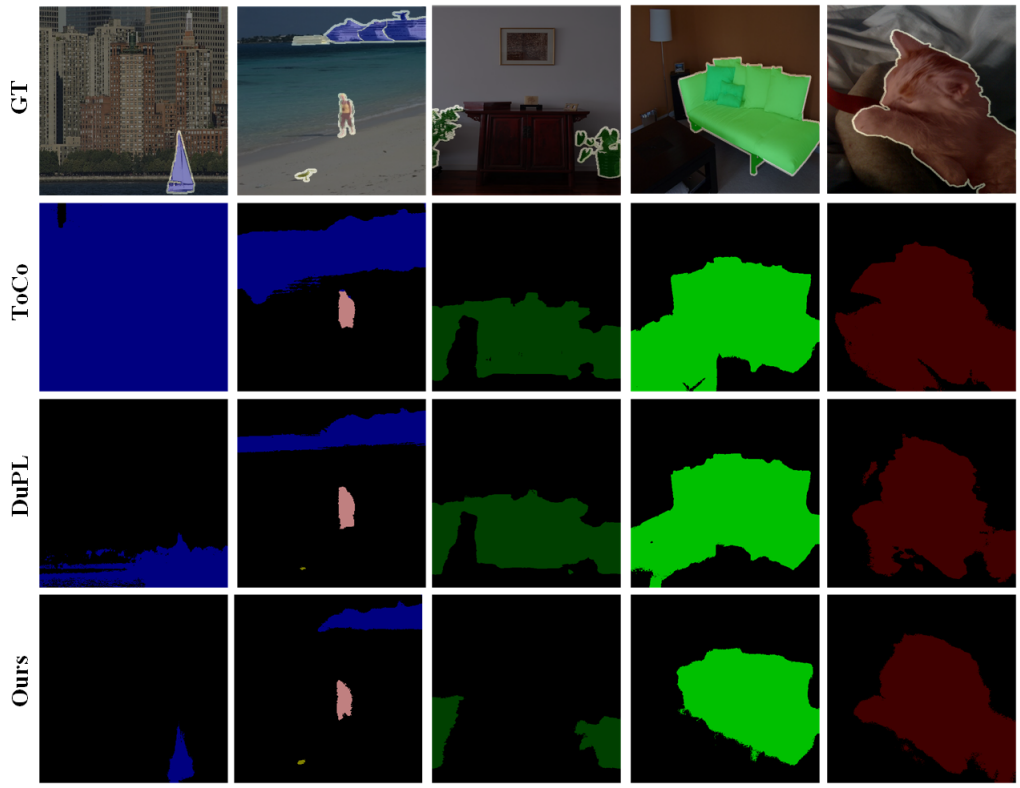

Figure 3. More visualization results of Segmentation on the PASCAL VOC 2012.

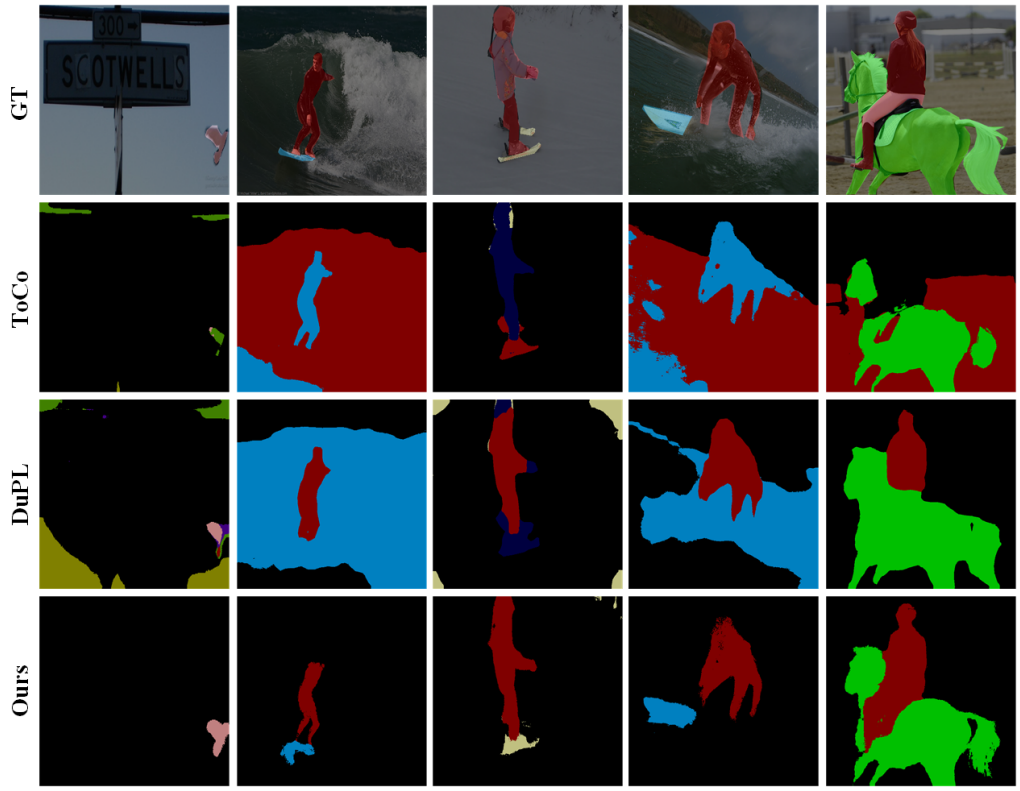

Figure 4. More visualization results of Segmentation on the MS COCO.