1. More Technical Details

1.1. Details of Backbone

We use the MiT-B1 proposed in SegFormer [5] as the backbone, which is a more friendly backbone for image segmentation tasks than the vanilla ViT [4]. SegFormer uses Overlapped Patch Merging layers with different strides to produce multi-scale feature maps. As shown in Fig. 1, in SegFormer the feature of Stage #4 is $\frac{h}{32} \times \frac{w}{32}$. To obtain the initial pseudo labels (CAM) with higher resolution, we change the stride of the last patch merging layer from 2 to 1, leading to the feature maps with the size of $\frac{h}{16} \times \frac{w}{16}$.

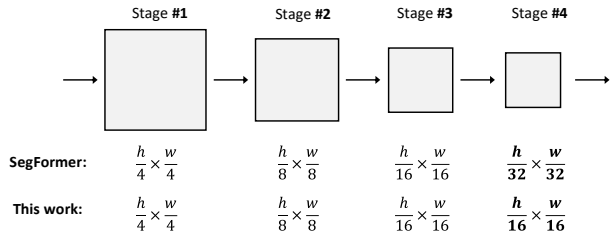

Figure 1. The size of feature maps of different stags.

In practice, to produce the semantic affinity prediction, we use the multi-head self-attention (MHSA) matrices extracted from the last stage, which could capture the high-level semantic affinity. The MHSA matrices are concatenated to form $S \in \mathbb{R}^{\frac{hw}{256} \times \frac{hw}{256} \times nk}$ and prediction the semantic affinity, where n and k are the number of Transformer blocks and heads in each block, respectively.

1.2. Mask for Affinity Loss

Inspired by [1, 2], when computing affinity loss, we only consider the situation that pixel pairs are in the same local window with the radius of r, and disregard their affinity if the distance is too far. Specifically, given a pixel (i, j), if pixel (k, l) is the same window with (i, j), their affinity is computed; otherwise, their affinity is ignored. Unlike [1, 2], which extract pixel pairs when computing affinity loss, we efficiently implemented by applying a mask. The conceptual illustration of this strategy and an example mask is presented in Fig. 2.

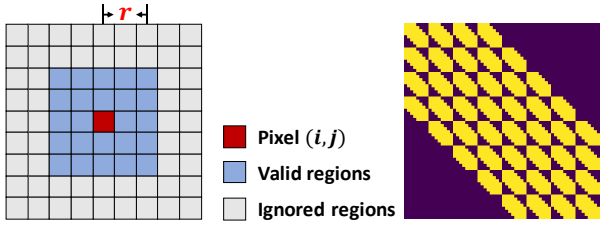

Figure 2. *Left*: Illustration of the valid pixel pairs. *Right*: Example mask for computing the affinity loss.

2. More Experimental Results

2.1. Hyper-parameters

Affinity from Attention. In Tab. 1, we present segmentation results on the PASCAL VOC val set with different radius r of the local window size when computing the affinity loss. Intuitively, a small r can not provide enough affinity pairs while a large r may not ensure the reliability of distant affinity pairs. As shown in Tab. 1, $r = 8$ is a proper choice.

Table 1. Impact of the radius r when computing the affinity loss. The results are evaluated on the val set of PASCAL VOC 2012.

radius r	2	4	8	12	16
val	62.4	62.7	**63.8**	61.5	59.4

Pixel-Adaptive Refinement. In Tab. 2, we report the impact of different configurations of the proposed Pixel-Adaptive Refinement, including the dilation rates, position kernel, and the number of iteration. Tab. 2 shows that for the same dilation rates, our PAR remarkably outperforms PAMR [3], demonstrating the necessity of the position kernel.

Table 2. Ablation of the dilation rates, position kernel and number of iteration of the proposed PAR. The results are evaluated on the $train$ set of PASCAL VOC 2012 in mIoU (%).

	Dilations						κ_{pos}	Iter	$train$
	1	2	4	8	12	24			
CAM									48.2
PAMR[3]	✓	✓	✓	✓	✓	✓			51.4
CRF									**54.5**
PAR	✓	✓	✓				✓	15	48.8
	✓	✓	✓	✓			✓	15	49.9
	✓	✓	✓	✓	✓		✓	15	51.3
	✓	✓	✓	✓	✓	✓		15	51.5
	✓	✓	✓	✓	✓	✓	✓	15	**52.9**
	✓	✓	✓	✓	✓	✓	✓	20	**52.9**

Tab. 3 presents the impact of the weights factors of PAR. For simplicity, we set $w_1 = w_2$. Tab. 3 shows $w_1 = 0.3, w_2 = 0.3, w_3 = 0.01$ is a favorable choice.

Weight Factors. We present the segmentation results on the PASCAL VOC val set with different weight factors of loss terms in Tab. 4. $\lambda_1 = 0.1, \lambda_2 = 0.2, \lambda_2 = 0.01$ is a preferred choice for our framework.

Background Scores We investigate the impact of the background scores (β_l, β_h) to filter the pseudo labels to the reliable foreground, background, and uncertain regions. Intuitively, large β_h and small β_l could produce more reliable pseudo labels but reduce the number of valid labels. On the contrary, small β_h and large β_l will introduce noise to the

Table 3. Ablation of weight factors of the proposed PAR. The results are evaluated on the *train* set of PASCAL VOC 2012.

		w_3			
		0.005	0.01	0.02	0.03
w_1 & w_2	0.1	51.9	51.7	50.1	–
	0.3	52.8	**52.9**	51.4	48.4
	0.5	51.9	52.5	51.3	48.3
	0.7	–	51.6	50.9	48.0

Table 4. Impact of the weights of loss terms. The results are evaluated on the *val* set of PASCAL VOC 2012.

	λ_1	λ_2	λ_3	*val*
Default	0.1	0.1	0.01	**63.8**
	0.05			62.8
	0.2			61.6
	0.5			57.8
		0.05		63.4
		0.2		61.7
		0.5		58.7
			0.005	62.4
			0.02	62.3
			0.05	61.5

Table 5. Impact of the background scores β_h, β_l. The results are evaluated on the *val* set of PASCAL VOC 2012.

	β_h	β_l	*val*
	0.65	0.25	60.7
	0.6	0.3	62.5
Default	0.55	0.35	**63.8**
	0.5	0.4	62.9
	0.45	0.45	60.5

pseudo labels. Note that the average value of β_h and β_l is always 0.45, which is the preferred background score for generated CAM in our preliminary experiments.

2.2. More Quantitative Results

We present the per-category segmentation results on PASCAL VOC *val* set in Tab 6. Our method achieves the best results for most categories. The results on *test* set are available at the official PASCAL VOC evaluation website[1].

2.3. More Qualitative Results

We present more qualitative results as follows.

[1] http://host.robots.ox.ac.uk:8080/anonymous/GHJIIH.html

Table 6. Evaluation and comparison of the semantic segmentation results in mIoU on the *val* set.

	RRM[6]	1Stage [3]	AA&LR [7]	**Ours**
bkg	87.9	88.7	88.4	**89.9**
aero	75.9	70.4	76.3	**79.5**
bicycle	31.7	**35.1**	33.8	31.2
bird	78.3	75.7	79.9	**80.7**
boat	54.6	51.9	34.2	**67.2**
bottle	62.2	65.8	**68.2**	61.9
bus	80.5	71.9	75.8	**81.4**
car	73.7	64.2	**74.8**	65.4
cat	71.2	81.1	82.0	**82.3**
chair	30.5	30.8	**31.8**	28.7
cow	67.4	73.3	68.7	**83.4**
table	40.9	28.1	**47.4**	41.6
dog	71.8	81.6	79.1	**82.2**
horse	66.2	69.1	68.5	**75.9**
motor	70.3	62.6	**71.4**	70.2
person	72.6	74.8	**80.0**	69.4
plant	49.0	48.6	50.3	**53.0**
sheep	70.7	71.0	76.5	**85.9**
sofa	38.4	40.1	43.0	**44.1**
train	62.7	**68.5**	55.5	64.2
tv	58.4	**64.3**	58.5	50.9
mIOU	62.6	62.7	63.9	**66.0**

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*, pages 2209–2218, 2019. 1

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, pages 4981–4990, 2018. 1

[3] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *CVPR*, pages 4253–4262, 2020. 1, 2, 6

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 1

[5] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021. 1

[6] Bingfeng Zhang, Jimin Xiao, Yunchao Wei, Mingjie Sun, and Kaizhu Huang. Reliability does matter: An end-to-end weakly supervised semantic segmentation approach. In *AAAI*, number 07, pages 12765–12772, 2020. 2

[7] Xiangrong Zhang, Zelin Peng, Peng Zhu, Tianyang Zhang, Chen Li, Huiyu Zhou, and Licheng Jiao. Adaptive affinity loss and erroneous pseudo-label refinement for weakly supervised semantic segmentation. In *ACM MM*, 2021. 2

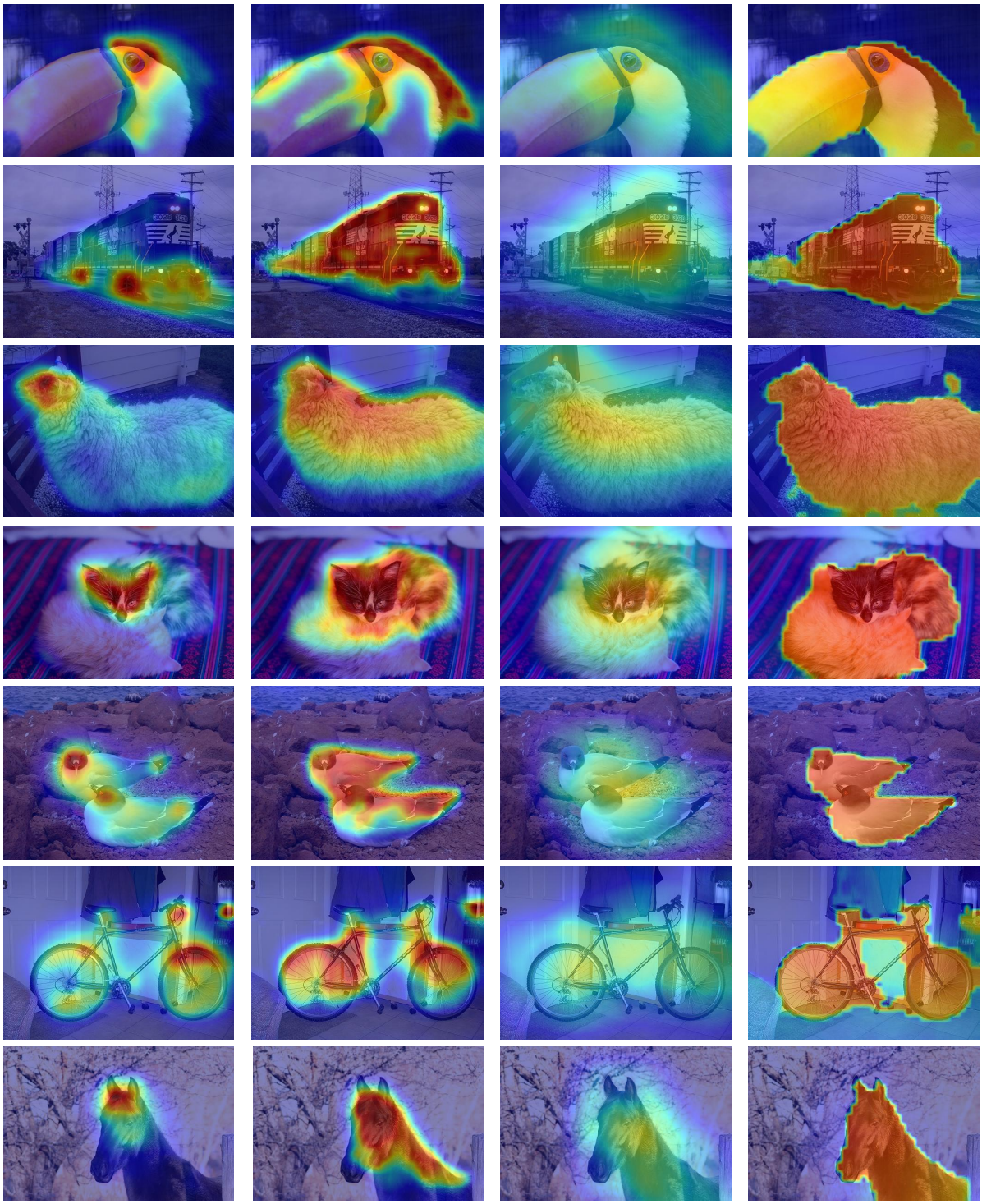

Figure 3. CAM generated with (a) Transformers activates more integral regions than (b) CNN. Refining CAM with (c) coarse MHSA doesn't work well, while (d) the learned affinity could remarkably improve the generated CAM.

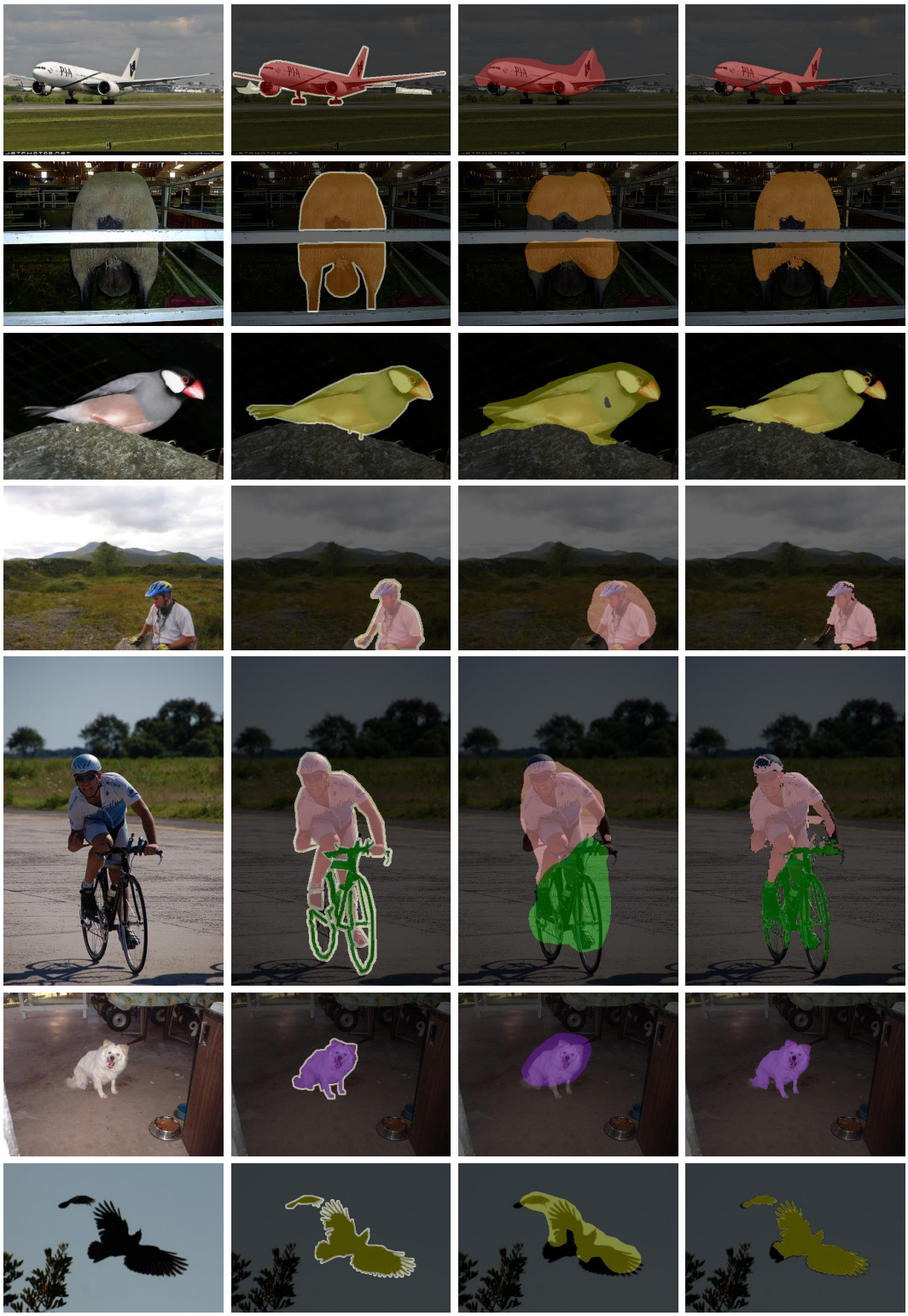

Figure 4. Improvements of the proposed pixel-adaptive refinement (PAR) module on the pseudo labels. The pseudo labels are generated with CAM and Transformer baseline. The proposed PAR could effectively dampen the falsely activated regions and ensure the alignment with low-level image appearance.

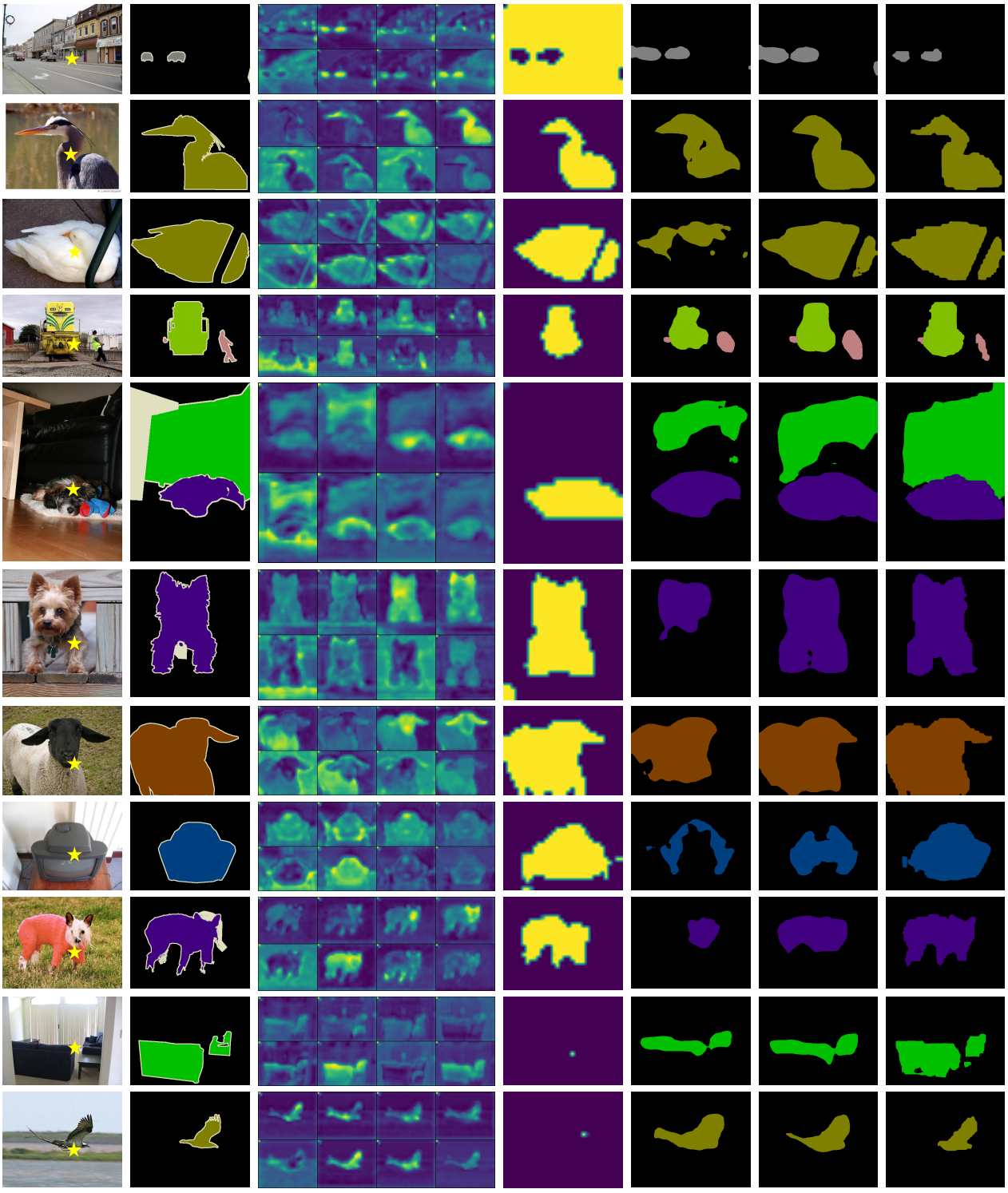

Figure 5. Visualization of the MHSA maps, learned affinity maps, and generated pseudo labels for segmentation. "★" denotes the query point to visualize the attention and affinity maps. The pseudo labels are generated with our model without AFA module (*w/o* AFA), with AFA module but no random walk propagation (AFA *w/o prop.*) and with full AFA module (with AFA). For the generated pseudo labels, the AFA module brings notable visual improvements. The affinity propagation process further diffuses the regions with high semantic affinity and dampens the regions with low affinity.

Figure 6. Semantic segmentation results on PASCAL VOC *val* (left) and MS COCO *val* set (right). Our method outperforms 1Stage [3] and is comparable with ground truth labels.

Figure 7. Visualization of the MHSA maps extracted from model without and with our AFA. "★" denotes the query point. Our AFA could help the MHSA to capture better semantic affinity.

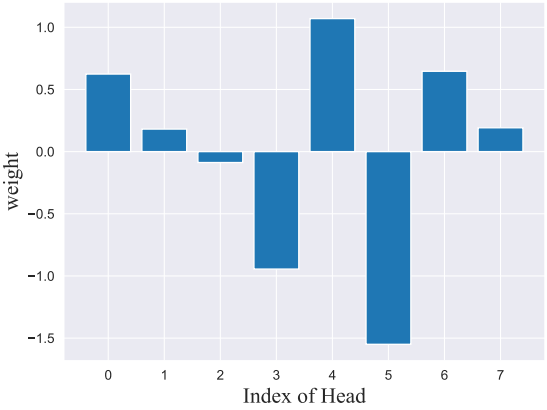

Figure 8. The learned weights of each head of self-attention in the AFA module. Here we only present the 8 heads of the last Transformer block. The MHSA matrices do not contribute equally to semantic affinity. Some self-attention matrices (head #2, head #3, and head #5) contribute negatively to semantic affinity. The learned weights suggest applying MHSA directly as semantic affinity is not beneficial for the pseudo labels.