Multi-class Token Transformer for Weakly Supervised Semantic Segmentation

-Supplementary Material-

A. Implementation details

A.1. Training and testing of MCTformer

To integrate the CAM module into the proposed MCTformer, we used a convolutional layer with C kernels of 3×3, a stride of 1, and a padding of 1, where C is the number of classes. We used the AdamW optimizer to train MCTformer with a batch size of 64 and an initial learning rate of 5×10^{-4}. The number of training epochs was set to 60. At test time, we generated transformer attention maps by fusing attentions from multiple layers. More specifically, to generate the class-specific object localization maps, we aggregated the class-to-patch transformer attention maps from the last three layers, as detailed in Figure 6 in Section 4.3. To generate the patch-level pairwise affinity, we aggregated the patch-to-patch transformer attention maps from all the twelve transformer layers, as the pairwise affinity is class-agnostic and different transformer blocks can learn different-level similarities between tokens. By aggregating all patch-to-patch attentions, it can thus produce a more informative affinity map. For the aggregation method, we followed [4] to first average attention maps from all the heads in each transformer layer, and then added up the averaged attention maps from all the selected layers, and finally performed a normalization to output a target attention map. For the evaluation of the generated class-specific object localization maps, we followed [6, 10] to report optimal results obtained when applying a range of thresholds to determine background pixels.

A.2. Training and testing for semantic segmentation

Following prior works [1, 10, 12, 13], we used ResNet38 based DeepLab-V1 as the segmentation model. For data augmentation, we used random scaling with a factor of ± 0.3, random horizontal flipping, random cropping to size 321×321. The polynomial learning rate decay was chosen with an initial learning rate of 7×10^{-4} and a power of 0.9. We used the stochastic gradient descent (SGD) optimizer to train the segmentation network for 30 epochs with a batch size of 4. At test time, we used multi-scale testing, *i.e.*, using inputs of multiple scales (0.5, 0.75, 1.0, 1.25, 1.5), and max-pooling for aggregating outputs, and the CRF with the default hyper-parameters suggested in [3] for post-processing.

B. Additional quantitative results

We reported the per-class IoU results on both the *val* and *test* sets of PASCAL VOC, and the *val* set of MS COCO in Table 1 and Table 2, respectively. These results show that the proposed MCTformer outperforms other state-of-the-art methods on most object categories, which demonstrates the superior performance of the proposed method.

C. Additional qualitative results

More qualitative segmentation results on the PASCAL VOC and MS COCO *val* sets are presented in Figure 1 and Figure 2. We can observe that the segmentation model trained with the pseudo labels generated by the proposed method produces satisfactory segmentation results. The model can segment large-scale objects with clear boundaries, and segment small-scale objects with fine-grained details in various indoor and outdoor scenes.

Figure 3 shows examples of the learned affinity maps of selected points (marked by the green crosses) in the input images, the generated transformer attention maps and their refined results by using the learned patch-to-patch transformer attention of the proposed MCTformer-V1. The learned affinity map (see Figure 3b) represents the similarity of the selected patch to all patches in the image. The cold (blue) to warm (red) colors denote low to high attention scores. As shown in the first row of Figure 3b, the affinity map highlights almost the entire object region for the class "dog". Although the object patch is not activated in the original transformer attention map (marked by the red square in the first row of Figure 3c), the learned affinity propagates the activations from similar regions to this patch, thus increasing its activation scores (marked by the red square in Figure 3d). More examples of the class-specific transformer attention maps and corresponding refined results by using the affinity maps from the proposed MCTformer-V1 are presented in Figure 4. These results show that the proposed MCTformer-V1 can effectively generate class-specific object localization maps from the transformer attentions. In addition, the patch-to-patch transformer attention of MCTformer-V1 is used as affinity maps, which can not only activate non-discriminative regions, but also refine object localization maps with noise filtering to produce more accurate object maps and boundaries.

More examples of the generated class-specific object localization maps from MCTformer-V2 on PASCAL VOC and MS COCO are presented in Figure 5 and Figure 6, respectively. Figure 5b shows the PatchCAM maps that are extracted from the transformed patch tokens. The global receptive field of the transformer is beneficial for CAM to localize a full context of large-scale objects (*e.g.*, the "plane" and the "train" in the third and fourth rows), while it leads to

over-activated localization maps for small-scale or irregular objects, such as the "bird" and the "plant" in the first and the last rows of Figure 5. In contrast, as shown in Figure 5c, the transformer attention usually allocates small values evenly to large-scale objects, due to the self-attention mechanism that all attention values of a class token are summed up to one. For small-scale or slim objects such as the "bird" in the first row of Figure 5, the proposed transformer attention can generate object localization maps with clear boundaries. The fusion of these two complementary maps, *i.e.*, the PatchCAM maps and the class-specific transformer attention maps, leads to significantly improved class-specific object localization maps with highly activated object regions and largely suppressed noise (Figure 5d and Figure 6d). Applying the patch-level pairwise affinity on the fused maps from these two can generate further refined object localization maps (Figure 5e and Figure 6e).

Table 1. Per-class performance comparison with the state-of-the-art WSSS methods in terms of IoUs (%) on PASCAL VOC. * denotes without post-processing.

	bkg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mIoU
Results on the *val* set:																						
SEAM (CVPR20) [10]	88.8	68.5	33.3	85.7	40.4	67.3	78.9	76.3	81.9	29.1	75.5	48.1	79.9	73.8	71.4	75.2	48.9	79.8	40.9	58.2	53.0	64.5
BEC (ECCV20) [2]	88.9	74.1	29.8	81.3	53.3	69.9	**89.4**	**79.8**	84.2	27.9	76.9	46.6	78.8	75.9	72.2	70.4	50.8	79.4	39.9	65.3	44.8	65.7
AdvCAM (CVPR21) [6]	90.0	79.8	34.1	82.6	**63.3**	70.5	**89.4**	76.0	87.3	31.4	81.3	33.1	82.5	80.8	74.0	72.9	50.3	82.3	42.2	**74.1**	52.9	68.1
ECS-Net (ICCV21) [9]	89.8	68.4	33.4	85.6	48.6	72.2	87.4	78.1	86.8	**33.0**	77.5	41.6	81.7	76.9	75.4	75.6	46.2	80.7	43.9	59.8	56.3	66.6
CDA (ICCV21) [8]	89.1	69.7	34.5	86.4	41.3	69.2	81.3	79.5	82.1	31.1	78.3	50.8	80.6	76.1	72.2	77.6	48.8	81.2	42.5	60.6	54.3	66.1
Zhang *et al.* (ICCV21) [13]	89.9	75.1	32.9	87.8	60.9	69.5	87.7	79.5	89.0	28.0	80.9	34.8	83.4	79.7	74.7	66.9	**56.5**	82.7	44.9	73.1	45.7	67.8
Kweon *et al.* (ICCV21) [5]	90.2	**82.9**	35.1	86.8	59.4	70.6	82.5	78.1	87.4	30.1	79.4	45.9	83.1	83.4	75.7	73.4	48.1	**89.3**	42.7	60.4	52.3	68.4
MCTformer* (Ours)	90.6	71.8	37.5	85.1	52.9	68.8	78.8	78.7	87.1	28.4	78.9	53.0	83.9	78.2	76.8	76.4	54.1	80.1	46.0	71.6	54.3	68.2
MCTformer (Ours)	**91.9**	78.3	**39.5**	**89.9**	55.9	**76.7**	81.8	79.0	**90.7**	32.6	**87.1**	**57.2**	**87.0**	**84.6**	**77.4**	**79.2**	55.1	89.2	**47.2**	70.4	**58.8**	**71.9**
Results on the *test* set:																						
AdvCAM (CVPR21) [6]	90.1	81.2	33.6	80.4	52.4	66.6	87.1	80.5	87.2	28.9	80.1	38.5	84.0	83.0	**79.5**	71.9	47.5	80.8	59.1	65.4	49.7	68.0
Zhang *et al.* (ICCV21) [13]	90.4	79.8	32.9	**85.8**	52.9	66.4	**87.2**	**81.4**	87.6	28.2	79.7	50.2	82.9	80.4	78.9	70.6	51.2	83.4	55.4	**68.5**	44.6	68.5
MCTformer* (Ours)	90.9	76.0	37.2	79.1	54.1	69.0	78.1	78.0	86.1	30.3	79.5	58.3	81.7	81.1	77.0	76.4	49.2	80.0	55.1	65.4	54.5	68.4
MCTformer (Ours)	**92.3**	**84.4**	**37.2**	82.8	**60.0**	**72.8**	78.0	79.0	**89.4**	**31.7**	**84.5**	**59.1**	**85.3**	**83.8**	79.2	**81.0**	**53.9**	**85.3**	**60.5**	65.7	**57.7**	**71.6**

Table 2. Per-class performance comparison with the state-of-the-art WSSS methods in terms of IoU(%) on the MS COCO *val* set.

Class	Luo *et al.* (AAAI20) [7]	AuxSegNet (ICCV21) [11]	MCTformer (Ours)	Class	Luo *et al.* (AAAI20) [7]	AuxSegNet (ICCV21) [11]	MCTformer (Ours)
background	73.9	82.0	**82.4**	wine class	27.2	**32.1**	27.0
person	48.7	**65.4**	62.6	cup	21.7	**29.3**	29.0
bicycle	45.0	43.0	**47.4**	fork	0.0	5.4	**13.9**
car	31.5	34.5	**47.2**	knife	0.9	1.4	**12.0**
motorcycle	59.1	**66.2**	63.7	spoon	0.0	1.4	**6.6**
airplane	26.9	60.3	**64.7**	bowl	7.6	19.5	**22.4**
bus	52.4	63.1	**64.5**	banana	52.0	46.9	**63.2**
train	42.4	57.3	**64.5**	apple	28.8	40.4	**44.4**
truck	36.9	38.9	**44.8**	sandwich	37.4	39.4	**39.7**
boat	23.5	30.1	**42.3**	orange	52.0	52.9	**63.0**
traffic light	13.3	40.4	**49.9**	broccoli	33.7	36.0	**51.2**
fire hydrant	45.1	72.7	**73.2**	carrot	29.0	13.9	**40.0**
stop sign	43.4	40.3	**76.6**	hot dog	38.8	46.1	**53.0**
parking meter	33.5	59.8	**64.4**	pizza	**69.8**	62.0	62.2
bench	26.3	16.0	**32.8**	donut	50.8	43.9	**55.7**
bird	29.9	61.0	**62.6**	cake	37.3	30.6	**47.9**
cat	62.1	68.6	**78.2**	chair	10.7	11.4	**22.8**
dog	57.5	66.9	**68.2**	couch	9.4	14.5	**35.0**
horse	40.7	55.6	**65.8**	potted plant	**21.8**	2.1	13.5
sheep	54.0	61.4	**70.1**	bed	34.6	20.5	**48.6**
cow	47.2	60.7	**68.3**	dining table	1.1	9.5	**12.9**
elephant	64.3	76.1	**81.6**	toilet	43.8	57.8	**63.1**
bear	58.9	73.0	**80.1**	tv	11.5	36.0	**47.9**
zebra	60.7	80.8	**83.0**	laptop	37.0	35.2	**49.5**
giraffe	45.1	71.6	**76.9**	mouse	0.0	**13.4**	**13.4**
backpack	0.0	11.3	**14.6**	remote	37.2	23.6	**41.9**
umbrella	46.1	35.0	**61.7**	keyboard	19.0	17.9	**49.8**
handbag	0.0	2.2	**4.5**	cellphone	38.1	49.9	**54.1**
tie	15.5	14.7	**25.2**	microwave	**43.4**	28.7	38.0
suitcase	43.6	31.7	**46.8**	oven	29.2	13.3	**29.9**
frisbee	23.2	1.0	**43.8**	toaster	0.0	0.0	0.0
skis	6.5	8.1	**12.8**	sink	**28.5**	21.0	28.0
snowboard	10.9	7.6	**31.4**	refrigerator	23.8	16.6	**40.1**
sports ball	0.6	**28.8**	9.2	book	26.3	8.7	**32.2**
kite	14.0	**27.3**	26.3	clock	13.4	34.4	**43.2**
baseball bat	0.0	**2.2**	0.9	vase	**27.1**	25.9	22.6
baseball globe	0.0	**1.3**	0.7	scissors	**37.0**	16.6	32.9
skateboard	7.6	**15.2**	7.8	teddy bear	58.9	47.3	**61.9**
surfboard	17.6	17.8	**46.5**	hair drier	0.0	0.0	0.0
tennis racket	38.1	**47.1**	1.4	toothbrush	11.1	1.4	**12.2**
bottle	28.4	**33.2**	31.1	**mIoU**	29.9	33.9	**42.0**

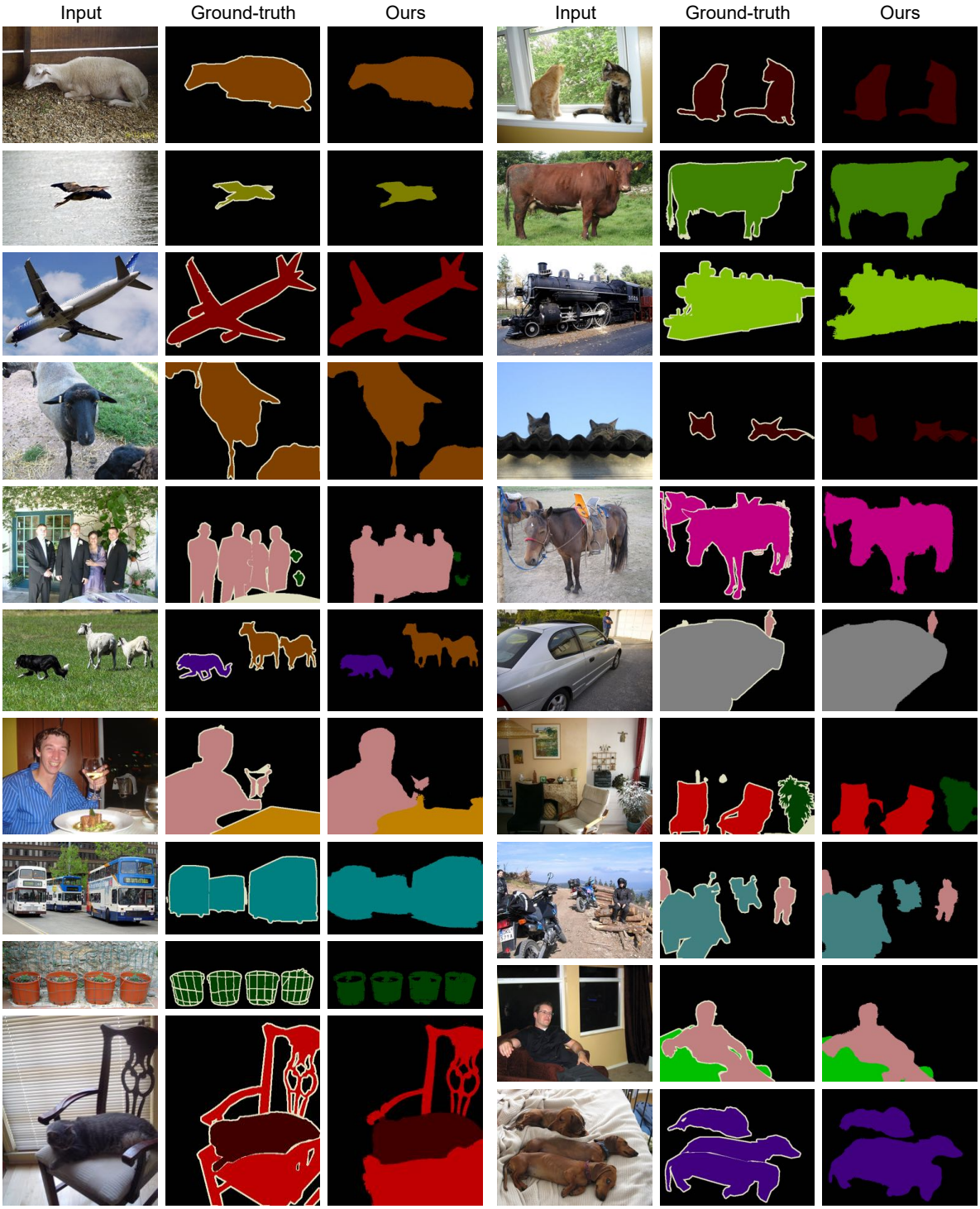

Figure 1. Qualitative segmentation results on the PASCAL VOC *val* set.

Figure 2. Qualitative segmentation results on the MS COCO *val* set.

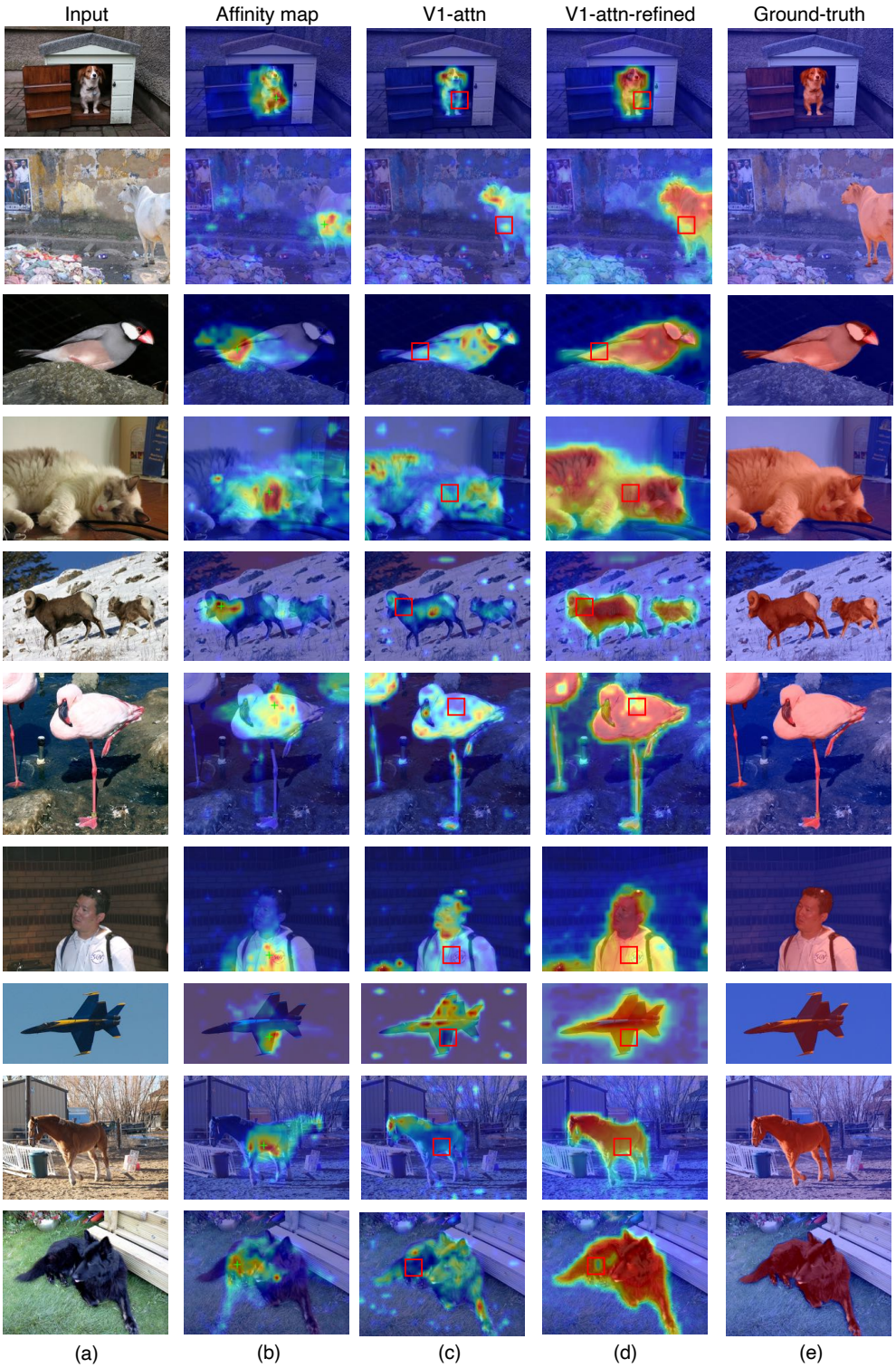

Figure 3. Visualization of the generated patch-level pairwise affinity from the proposed MCTformer-V1 on the PASCAL VOC *train* set. (a) Input; (b) Affinity map (the generated affinity maps for the points marked by the green crosses); (c) V1-attn (the generated transformer attention maps from MCTformer-V1, where the red squares denote the original attention scores for the corresponding points in (b)); (d) V1-attn-refined (the refined class-specific transformer attention maps, where the red squares denote the refined attention scores using the corresponding affinity maps of (b)); (e) Ground-truth.

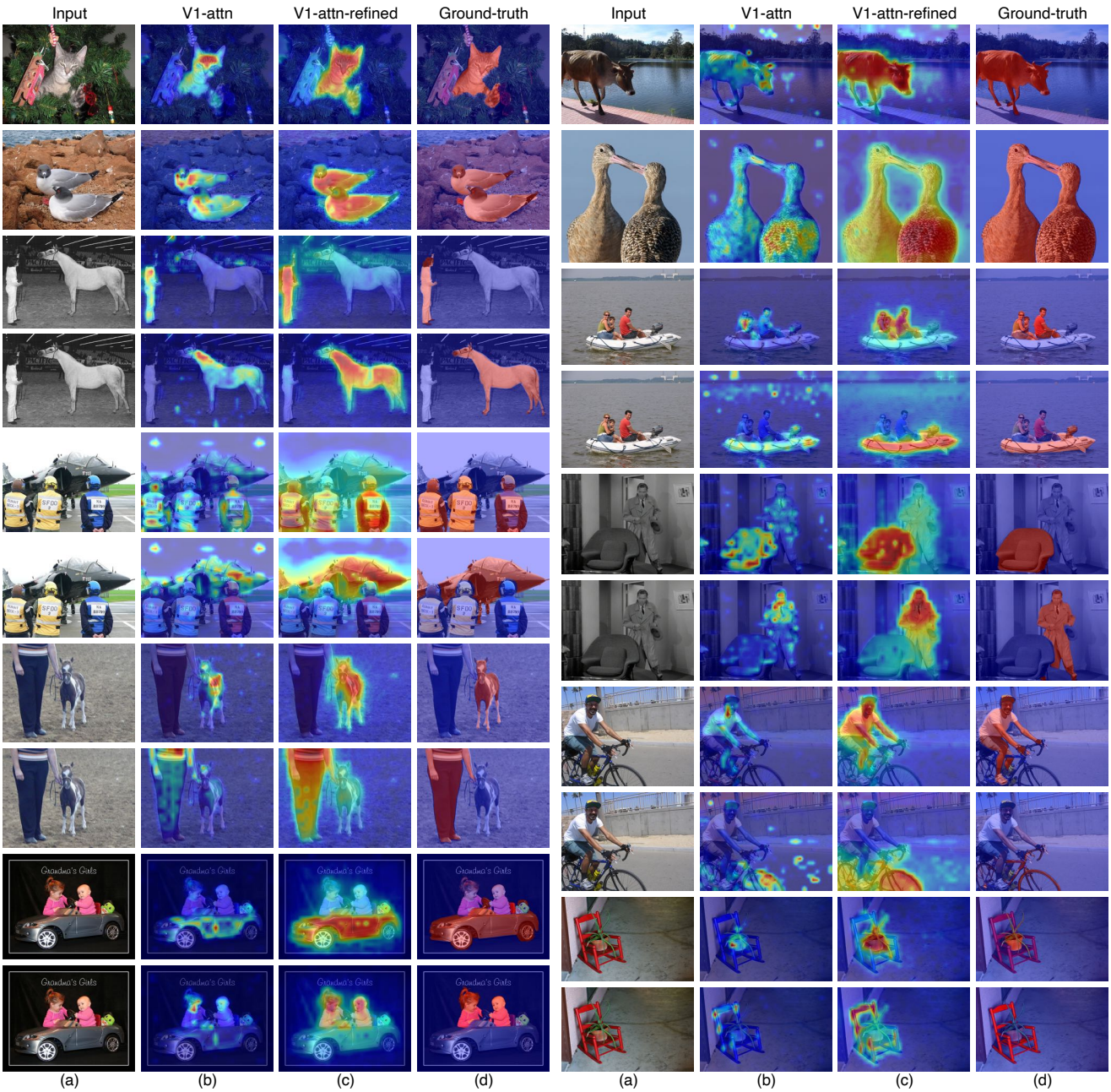

Figure 4. Visualization of the generated class-specific transformer attention maps and their refined results using the patch-level pairwise affinity from the proposed MCTformer-V1 on the PASCAL VOC *train* set. (a) Input; (b) V1-attn (the generated class-specific transformer attention maps from the proposed MCTformer-V1); (c) V1-attn-refined (the refined class-specific transformer attention maps using the patch-level pairwise affinity from the proposed MCTformer-V1); (d) Ground-truth.

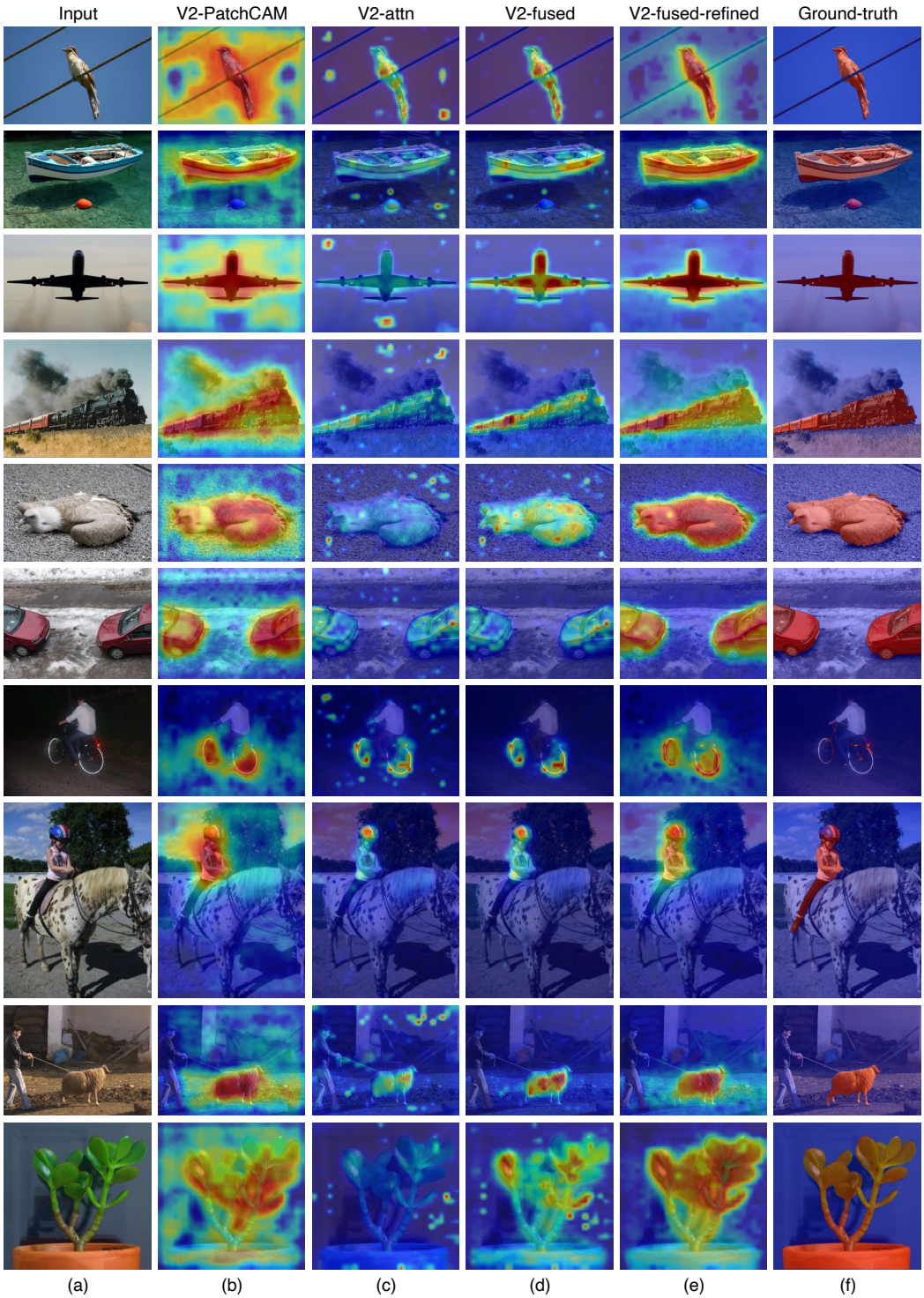

Figure 5. Visualization of the generated class-specific object localization maps from MCTformer-V2 on the PASCAL VOC *train* set. (a) Input; (b) V2-PatchCAM (the generated PatchCAM maps from MCTformer-V2); (c) V2-attn (the generated class-specific transformer attention maps from MCTformer-V2); (d) the fusion maps of (b) and (c); (e) the refined fusion maps by the patch-level pairwise affinity from MCTformer-V2; (f) Ground-truth.

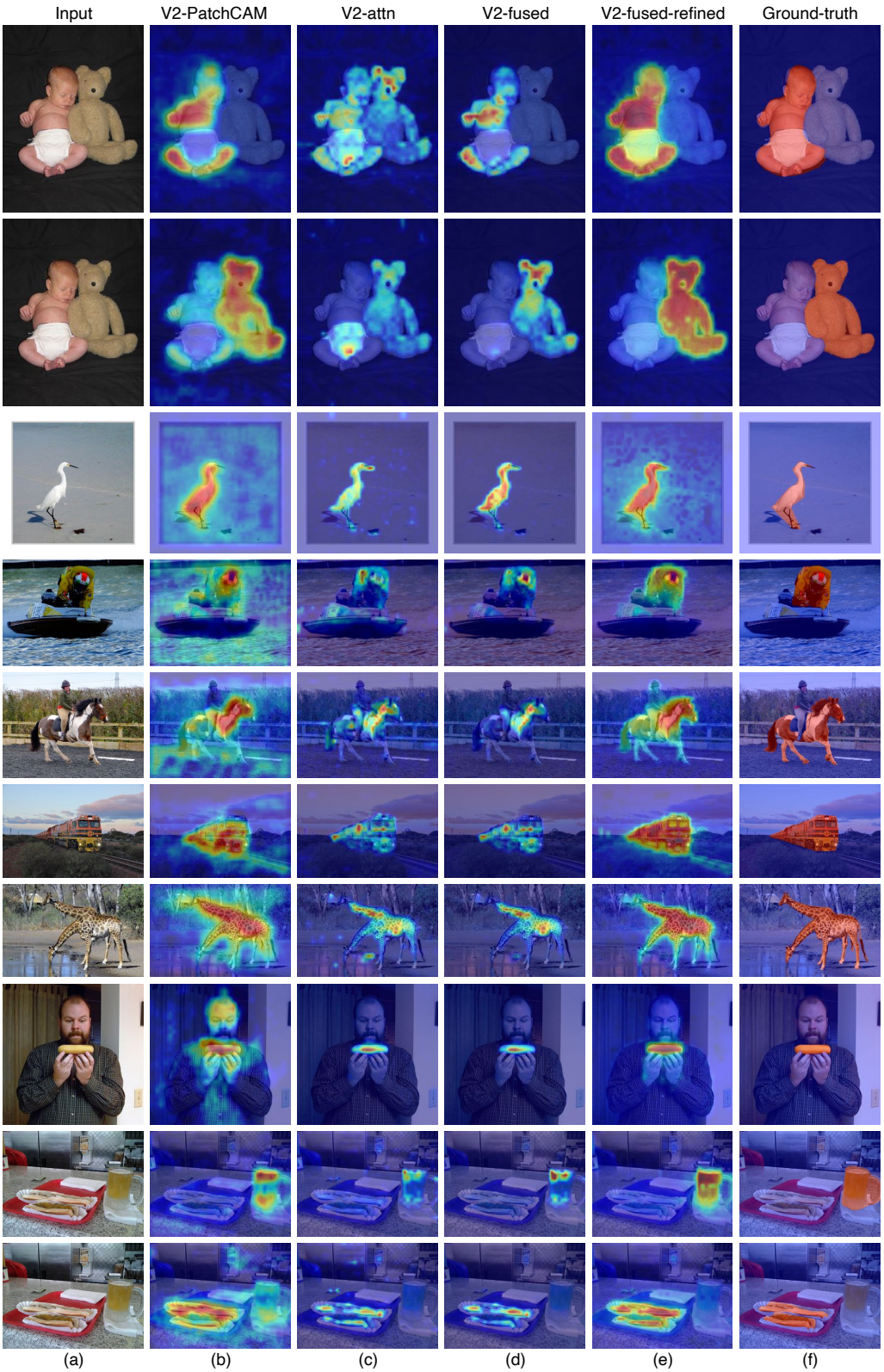

Figure 6. Visualization of the generated class-specific object localization maps from MCTformer-V2 on the MSCOCO *train* set. (a) Input; (b) V2-PatchCAM (the generated PatchCAM maps from MCTformer-V2); (c) V2-attn (the generated class-specific transformer attention maps from MCTformer-V2); (d) the fusion maps of (b) and (c); (e) the refined fusion maps by the patch-level pairwise affinity from MCTformer-V2; (f) Ground-truth.

References

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

[2] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *ECCV*, 2020. 3

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 1

[4] Wei Gao, Fang Wan, Xingjia Pan, Zhiliang Peng, Qi Tian, Zhenjun Han, Bolei Zhou, and Qixiang Ye. Ts-cam: Token semantic coupled attention map for weakly supervised object localization. In *ICCV*, 2021. 1

[5] Hyeokjun Kweon, Sung-Hoon Yoon, Hyeonseong Kim, Daehee Park, and Kuk-Jin Yoon. Unlocking the potential of ordinary classifier: Class-specific adversarial erasing framework for weakly supervised semantic segmentation. In *ICCV*, 2021. 3

[6] Jungbeom Lee, Eunji Kim, and Sungroh Yoon. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In *CVPR*, 2021. 1, 3

[7] Wenfeng Luo and Meng Yang. Learning saliency-free model with generic features for weakly-supervised semantic segmentation. In *AAAI*, 2020. 3

[8] Yukun Su, Ruizhou Sun, Guosheng Lin, and Qingyao Wu. Context decoupling augmentation for weakly supervised semantic segmentation. In *ICCV*, 2021. 3

[9] Kunyang Sun, Haoqing Shi, Zhengming Zhang, and Yongming Huang. Ecs-net: Improving weakly supervised semantic segmentation by using connections between class activation maps. In *ICCV*, 2021. 3

[10] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, 2020. 1, 3

[11] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, Ferdous Sohel, and Dan Xu. Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation. In *ICCV*, 2021. 3

[12] 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*, 2020. 1

[13] Fei Zhang, Chaochen Gu, Chenyue Zhang, and Yuchao Dai. Complementary patch for weakly supervised semantic segmentation. In *ICCV*, 2021. 1, 3