Weakly Supervised Semantic Segmentation via Adversarial Learning of Classifier and Reconstructor: *Supplementary Material*

Anonymous CVPR submission

Paper ID 5400

A. Implementation Details

A.1. Network Architecture

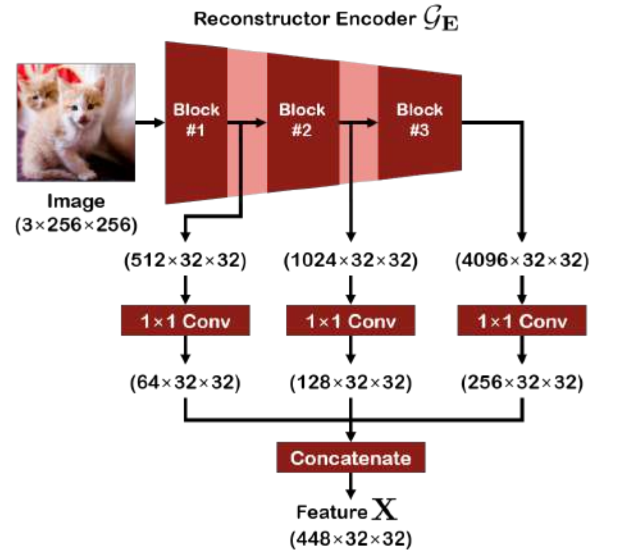

Figure A1. Visualization of the Reconstructor Encoder. We aggregate the feature maps from multiple different layers into a multi-scale feature map $\mathbf{X}$.

The proposed framework has two networks: a classifier $\mathcal{F}$ and a reconstructor $\mathcal{G}$. We employ the ResNet38 [8] as the backbone for the classifier, as in many other WSSS studies [2, 3, 5–7, 10, 11]. We add a 1×1 convolution layer to the backbone, as a classification head for acquiring CAMs.

We visualize the architecture of the reconstructor encoder $\mathcal{G}_E$ in Fig. A1. Similar to the classifier, we employ ResNet38 as a backbone for the encoder. Here, for better reconstruction capability, we aggregate the feature maps from multiple different layers into a multi-scale feature map. We add 1×1 convolution layers for integrating the feature maps. The feature $\mathbf{X}$ in the main paper denotes this multi-scale feature map, where the dimension d is set to 448. This design enables the encoder to extract both primitive details from low-level and context information from high-level.

Table A1. Dimensions of the output feature obtained by each block of our reconstructor decoder. For example, the output of the D1 block has a dimension of $64 \times 128 \times 128$. Note that the final output (of C block) is a reconstructed RGB image, and thereby has $3 \times 256 \times 256$ dimension.

Blocks	Input	D1	D2	D3	D4	D5	D6	D7	D8
Channel	448	64	128	256	512	512	512	512	512
Size	256	128	64	32	16	8	4	2	1
Blocks	-	U7	U6	U5	U4	U3	U2	U1	C
Channel	-	1024	1024	1024	1024	512	256	128	3
Size	-	2	4	8	16	32	64	128	256

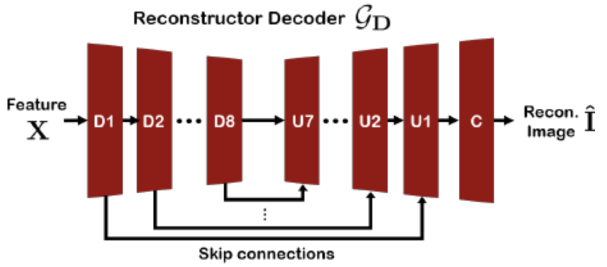

Figure A2. Visualization of the UNet-based Reconstructor Decoder. The decoder gets upscaled feature $\mathbf{X}$ and returns the reconstructed image $\hat{\mathbf{I}}$. $\mathbf{D}$, $\mathbf{U}$, and $\mathbf{C}$ denote the Downsample, Upsample, and Colorization blocks, respectively.

For the reconstructor decoder $\mathcal{G}_D$, we devise a UNet-based architecture, as shown in Fig. A2. Note that we upscale the feature $\mathbf{X}$ into the size of the input image, before passing it to the decoder. The decoder is composed of eight downsample (D) blocks, seven upsample (U) blocks, and one colorization (C) block. Each downsample block is a 4×4 convolutional layer with a stride of 2 followed by a normalization layer and leakyReLU (LReLU) activation. On the other hand, each upsample block has 4×4 transposed convolutional layer with a stride of 2. Similar to the downsample block, we also use the normalization layer and LReLU activation. Finally, the colorization block is composed of a $2\times$ bilinear upsample followed by a 1×1 convolutional layer for obtaining the output having 3 channels (RGB). For reproducibility, we provide the dimensions of the feature obtained by each block in Table A1.

Table A2. Ablation study on probability q. The mIoU performance on the PASCAL VOC 2012 train set is listed.

q	0.60	0.65	0.70	0.75	0.80	0.85	0.90
mIoU	59.7	59.6	59.6	60.1	60.3	60.3	60.0

A.2. Details on Stochastic Remnant Feeding (SRF)

As described in Section 4.2.2 of the main paper, we devise a Stochastic Remnant Feeding (SRF) technique. It aims to prevent the over-fitting of the reconstruction network, by feeding the synthetic remnants obtained by a stochastic grid. The stochastic grid is a random binary grid $\mathbf{g} \in [0, 1]^{h \times w}$ composed of $s \times s$ cells. Each $\frac{h}{s} \times \frac{w}{s}$ size cell has a value of 0 or 1, sampled from the Bernoulli distribution, which is independent and identical, having the probability of q.

For the number of cells s, we use an integer sampled from the uniform distribution in the range of 16 to 24, as $s \sim U_{[16,24]}$. The sampling is performed in every iteration. We have adjusted the range of sampling, however, no meaningful increase in performance is observed.

For the probability of the Bernoulli distribution, we set the initial value of q as 0.8. Note that it means that 80% of the grid has a value of 1, and the other 20% has 0. At the early stage of learning, the reconstructor is not very specialized in using the remnants for inferring one segment from the other. Therefore, we initially set q as a value high enough, which can be interpreted as giving an easy problem for the reconstructor to solve. Then, we decrease the probability as training proceeds (1% per epoch). Similar to curriculum learning, the reconstructor is trained to solve a more difficult problem as training proceeds and to better exploit the remnants for reconstructing one segment from the other segment, as we intended. We experimentally verified that using this strategy provides around 1% gain. We also provide the mIoU performance achieved by using the various different initial values for q, as in Table. A2. The results support the robustness of the proposed SRF strategy against the change of the probability q.

A.3. Settings for MS COCO dataset

Training CAMs Our framework is trained on MS COCO dataset for 3 epochs, which took a day with a single RTX 3090 ti. The weighting parameters λ_t^{RU} and λ_{nt}^{RU} are equally set to 0.2. For the λ_t^{CU} and λ_{nt}^{CU}, we set the values to 0.3 and 0.2, respectively. The learning rate and batch size for COCO are set to 0.005 and 8, respectively.

Training Semantic Segmentation For the training of the semantic segmentation model, we used a smaller learning rate (5×10^{-4}) than in training the CAMs. We trained the model with 30 epochs using the obtained pseudo-labels of 81k *train* set. The weight decay and batch size are set to 5×10^{-5} and 8, respectively.

B. Incorporating with Transformer

Recently, due to its remarkable representation capability, Vision Transformer (ViT) [4] is widely used in various computer vision tasks, including WSSS. For a fair comparison with the ViT-based method [9], we further incorporate the proposed method with the ViT.

Our original implementation is based on the conventional convolutional neural network (CNN). However, the proposed philosophy (*i.e.*, adversarial learning of classifier and reconstructor) does not have an explicit limitation on its choice of backbones. As in MCTformerV2 [9], we refine the CAM obtained by the proposed method, using attention between the patch tokens. Note that we use the same reconstructor with our CNN-based implementation. We provide the quantitative evaluation of this incorporated version, denoted as Ours (+ViT), in the main paper. This setting still outperforms its baseline (MCTformerV2 [9]) even when using the ViT backbone, supporting the superiority of the proposed method. We show some CAMs (Fig. A3) and semantic segmentation results (Fig. A4) of Ours (+ViT), comparing with those of the MCTformerV2.

C. Results on PASCAL VOC

CAMs In Fig. A5, we show the CAMs obtained by our framework, which were omitted from the main paper due to page limit. The image samples are from PASCAL VOC 2012 *train* set. As we can refer from Fig. A5, the CAMs from our proposed method are not only precise but also more evenly distributed.

Reconstructed Images In Fig. A6, we provide some samples of the reconstructed images obtained by our framework, in addition to the target/non-target CAMs and binary grid for SRF. In detail, we visualize the reconstructed images from RU phase ($\hat{\mathbf{I}}_t^{RU}$, $\hat{\mathbf{I}}_{nt}^{RU}$) and CU phase ($\hat{\mathbf{I}}_t^{CU}$, $\hat{\mathbf{I}}_{nt}^{CU}$). As we can observe $\hat{\mathbf{I}}_t^{RU}$, in RU phase, the reconstructor could reconstruct the non-target regions with the help of the remnants, and so on for $\hat{\mathbf{I}}_{nt}^{RU}$. On the other hand, in CU phase, we can observe that the reconstructor failed to reconstruct the original image corresponding to the non-target regions as shown in $\hat{\mathbf{I}}_t^{CU}$. Similarly, $\hat{\mathbf{I}}_{nt}^{CU}$ shows that the reconstructor failed to reconstruct the target region. The results imply that the classifier and the reconstructor in our framework play their role as we intended.

Semantic Segmentation By applying IRN [1] to CAMs as previous WSSS methods, we acquired high-quality pseudo-labels with SoTA performance. After training the semantic segmentation with the pseudo-labels, we also achieve a new SOTA in semantic segmentation stage with only image-level supervision. Semantic segmentation results on PASCAL VOC 2012 *val* set are shown in Fig. A7. The mIoU performance on *test* set is shown here [1].

[1] http://host.robots.ox.ac.uk:8080/anonymous/ZQKP1X.html

D. Results on MS COCO

CAMs To show the superiority of the proposed method, we also conduct experiments on MS COCO dataset. Since the MS COCO dataset contains more classes and smaller objects than PASCAL VOC, it is more difficult to get precise CAMs/pseudo-labels. However, as shown in Fig A8, CAMs from the proposed method are not only precise but also capture small details (see the last row of Fig. A8, *backpack*) well. Also, as shown in the 4-6th row of Fig. A8, the CAMs are mutually exclusive while having accurate boundaries.

Semantic Segmentation As in PASCAL VOC, we obtained pseudo-labels by applying IRN [1] to CAMs. The mIoU of the pseudo-labels is 48.1% on 81k *train* set. The semantic segmentation model trained with the pseudo-labels achieves a new SoTA with 45.3% mIoU on 40k *val* set. Semantic segmentation results are also shown in Fig. A9.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2209–2218, 2019. 2, 3

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4981–4990, 2018. 1

[3] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8991–9000, 2020. 1

[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. *arXiv preprint arXiv:2010.11929*, 2020. 2

[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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6994–7003, 2021. 1

[6] Wataru Shimoda and Keiji Yanai. Self-supervised difference detection for weakly-supervised semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5208–5217, 2019. 1

[7] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12275–12284, 2020. 1

[8] Zifeng Wu, Chunhua Shen, and Anton Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019. 1

[9] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4310–4319, 2022. 2, 4, 5

[10] Sung-Hoon Yoon, Hyeokjun Kweon, Jegyeong Cho, Shinjeong Kim, and Kuk-Jin Yoon. Adversarial erasing framework via triplet with gated pyramid pooling layer for weakly supervised semantic segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIX*, pages 326–344. Springer Nature Switzerland Cham, 2022. 1

[11] Bingfeng Zhang, Jimin Xiao, Yunchao Wei, Mingjie Sun, and Kaizhu Huang. Reliability does matter: An end-to-end weakly supervised semantic segmentation approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12765–12772, 2020. 1

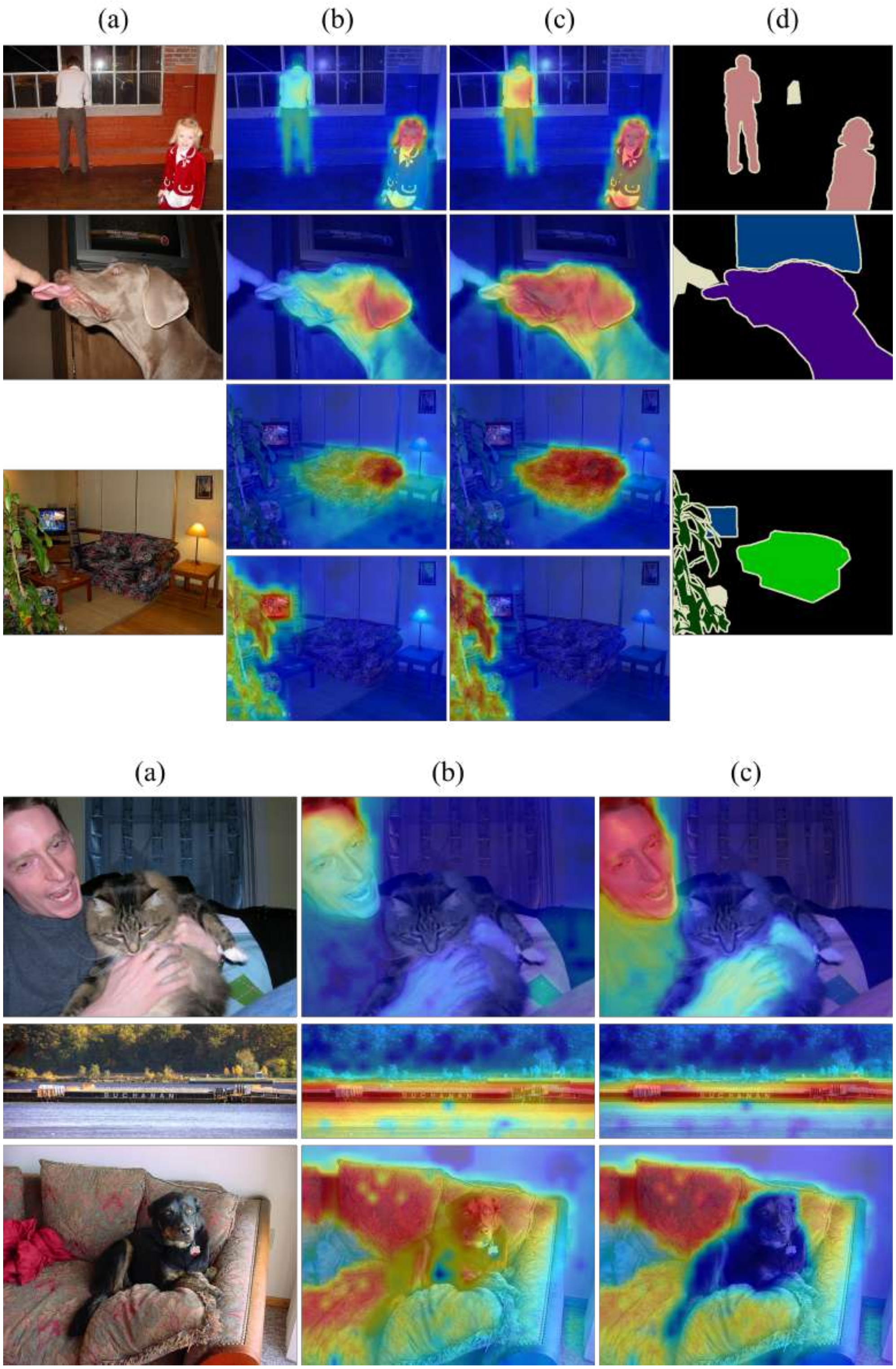

Figure A3. CAMs results on PASCAL VOC 2012 *train* and *trainaug* set. From (a) to (d): images, CAMs of MCTformerV2 [9], Our(+ViT) CAMs, and GTs (if exist). Note that the CAMs of the last raw correspond to the *sofa* class.

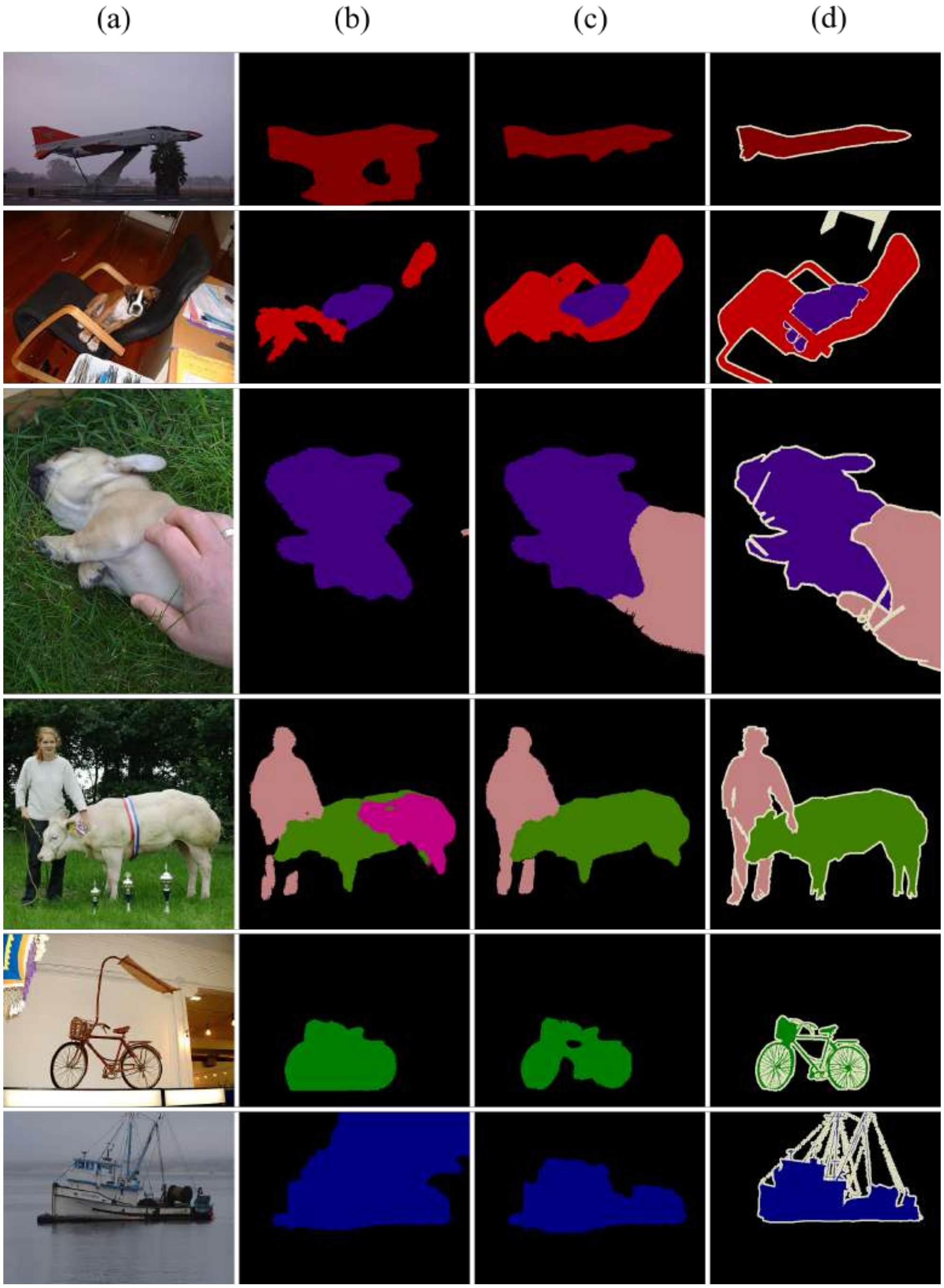

Figure A4. Semantic segmentation results on PASCAL VOC 2012 *train* set. From (a) to (d): images, results of MCTformerV2 [9], results of Our(+ViT), and GTs.

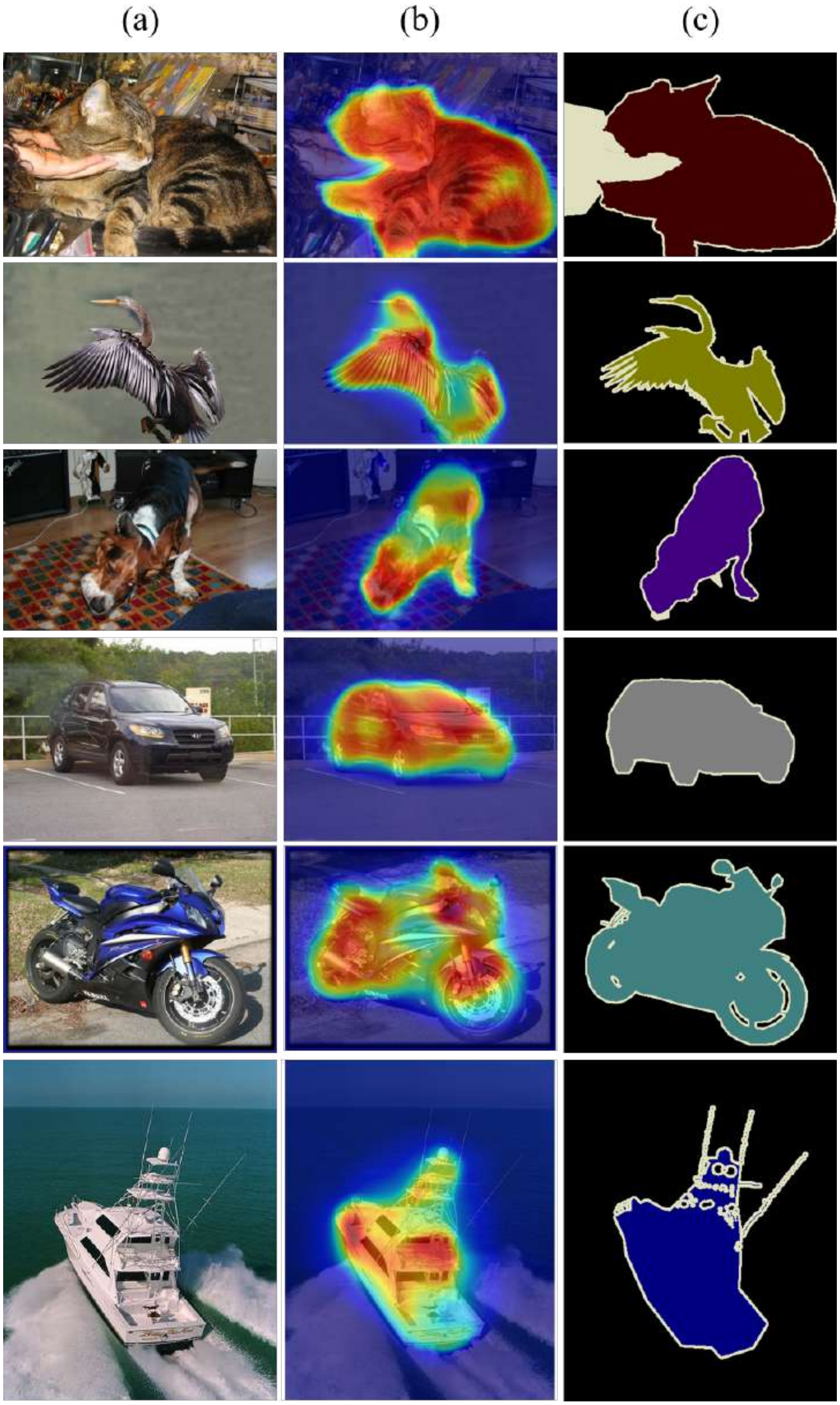

Figure A5. CAMs results on PASCAL VOC 2012 *train* set. From (a) to (c): images, our CAMs, and Ground-Truths (GTs).

$\mathbf{I}$	$\mathbf{A}_t$	$\hat{\mathbf{I}}_t^{RU}$	$\hat{\mathbf{I}}_t^{CU}$
g	$\mathbf{A}_{nt}$	$\hat{\mathbf{I}}_{nt}^{RU}$	$\hat{\mathbf{I}}_{nt}^{CU}$

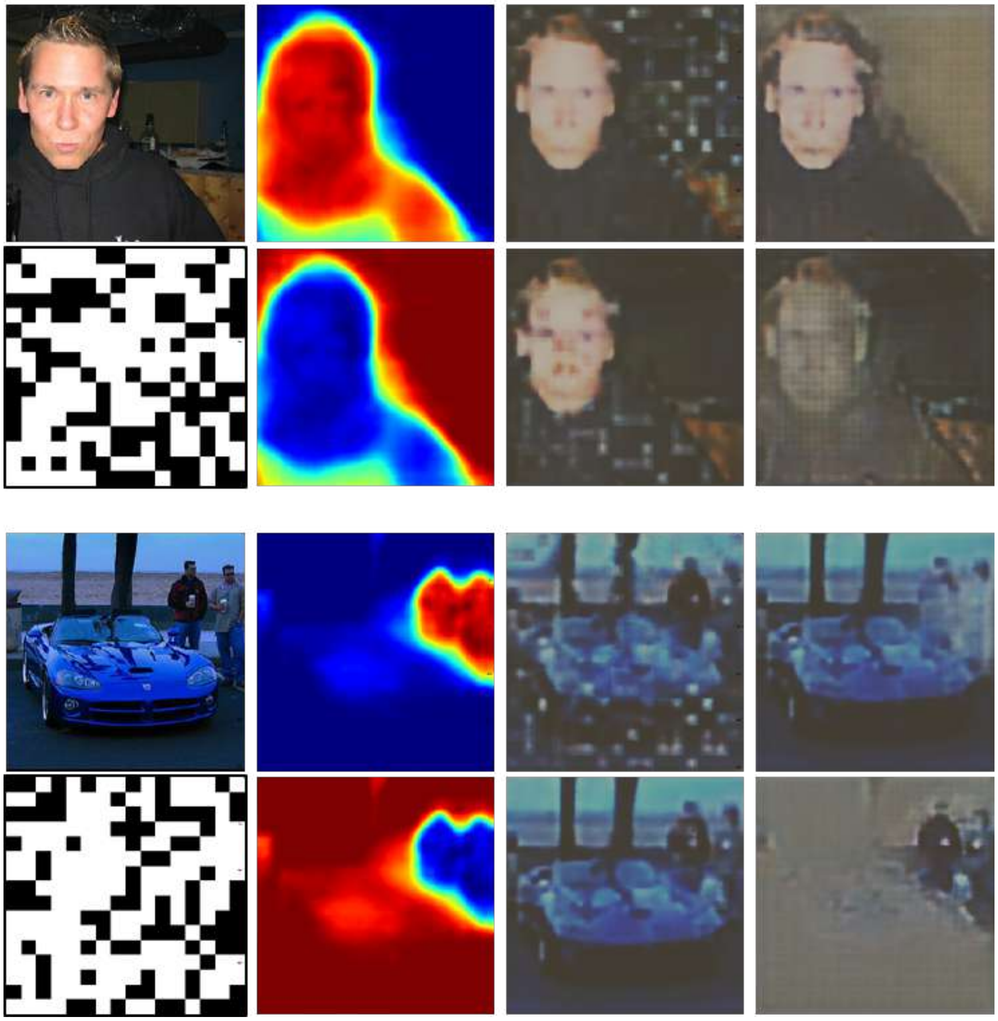

Figure A6. Samples of reconstructed results on PASCAL VOC 2012 *train* set. First row, from left to right: input image ($\mathbf{I}$), target CAM ($\mathbf{A}_t$), image reconstructed from target feature in RU phase ($\hat{\mathbf{I}}_t^{RU}$), and image reconstructed from target CAM in CU phase ($\hat{\mathbf{I}}_t^{CU}$). Second row, from left to right: binary grid (g) used for SRF, non-target CAM ($\mathbf{A}_{nt}$), image reconstructed from non-target feature in RU phase ($\hat{\mathbf{I}}_{nt}^{RU}$), and image reconstructed from non-target CAM in CU phase ($\hat{\mathbf{I}}_{nt}^{CU}$).

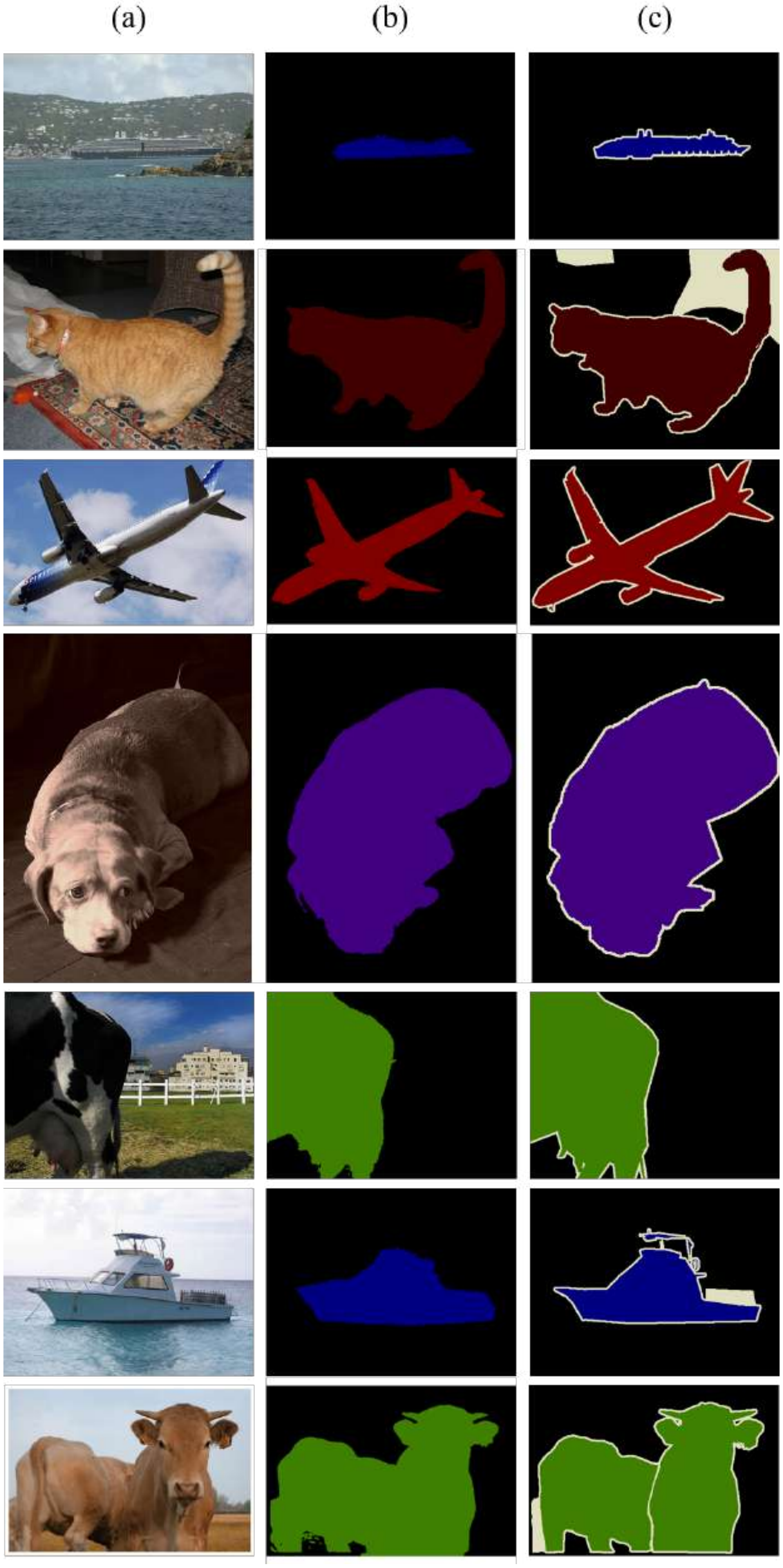

Figure A7. Semantic segmentation results on PASCAL VOC 2012 *validation* set. From (a) to (c): images, our Deeplab, GTs.

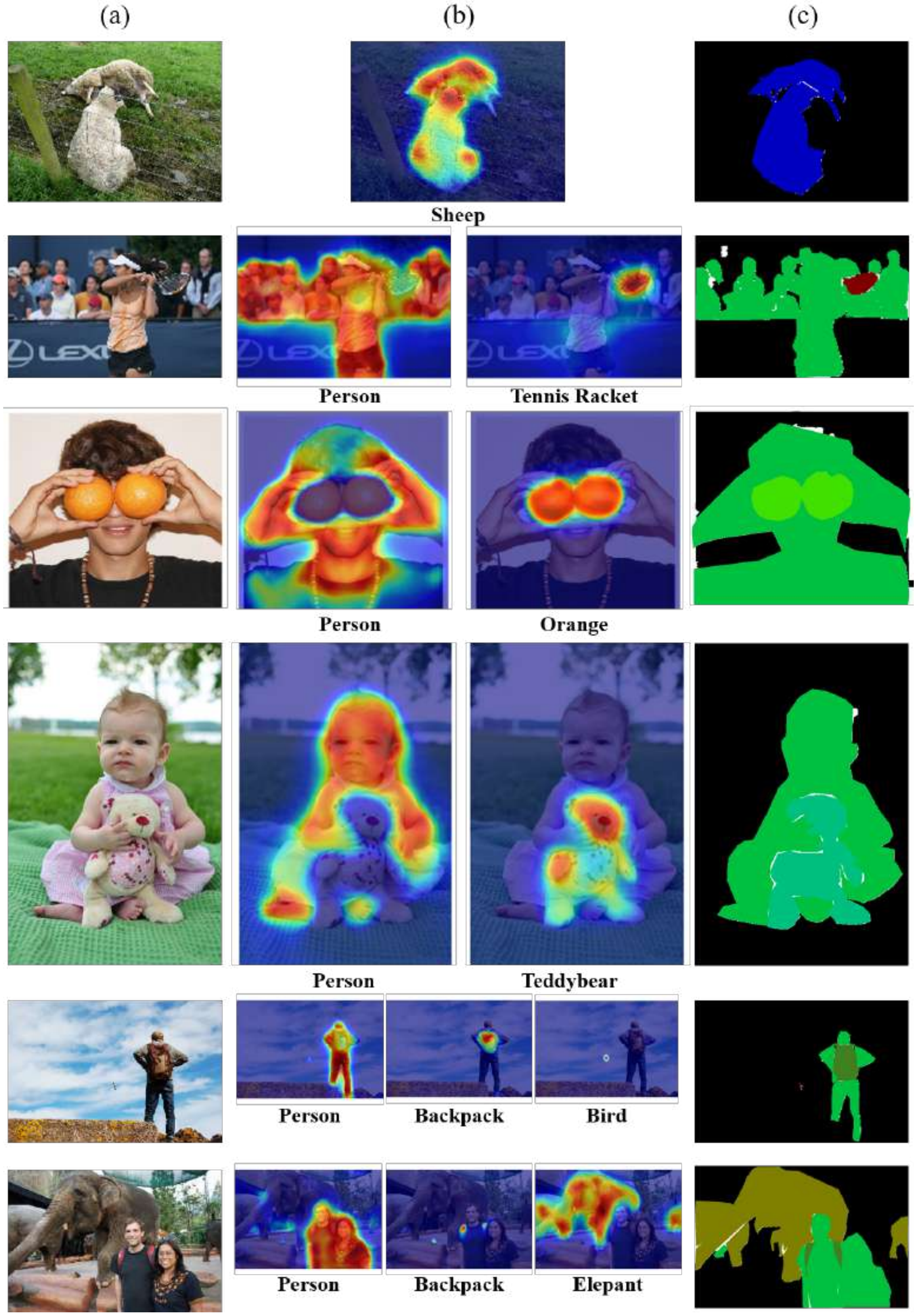

Figure A8. CAMs results on MS COCO *train* set. From (a) to (c): images, our CAMs, and Ground-Truths (GTs).

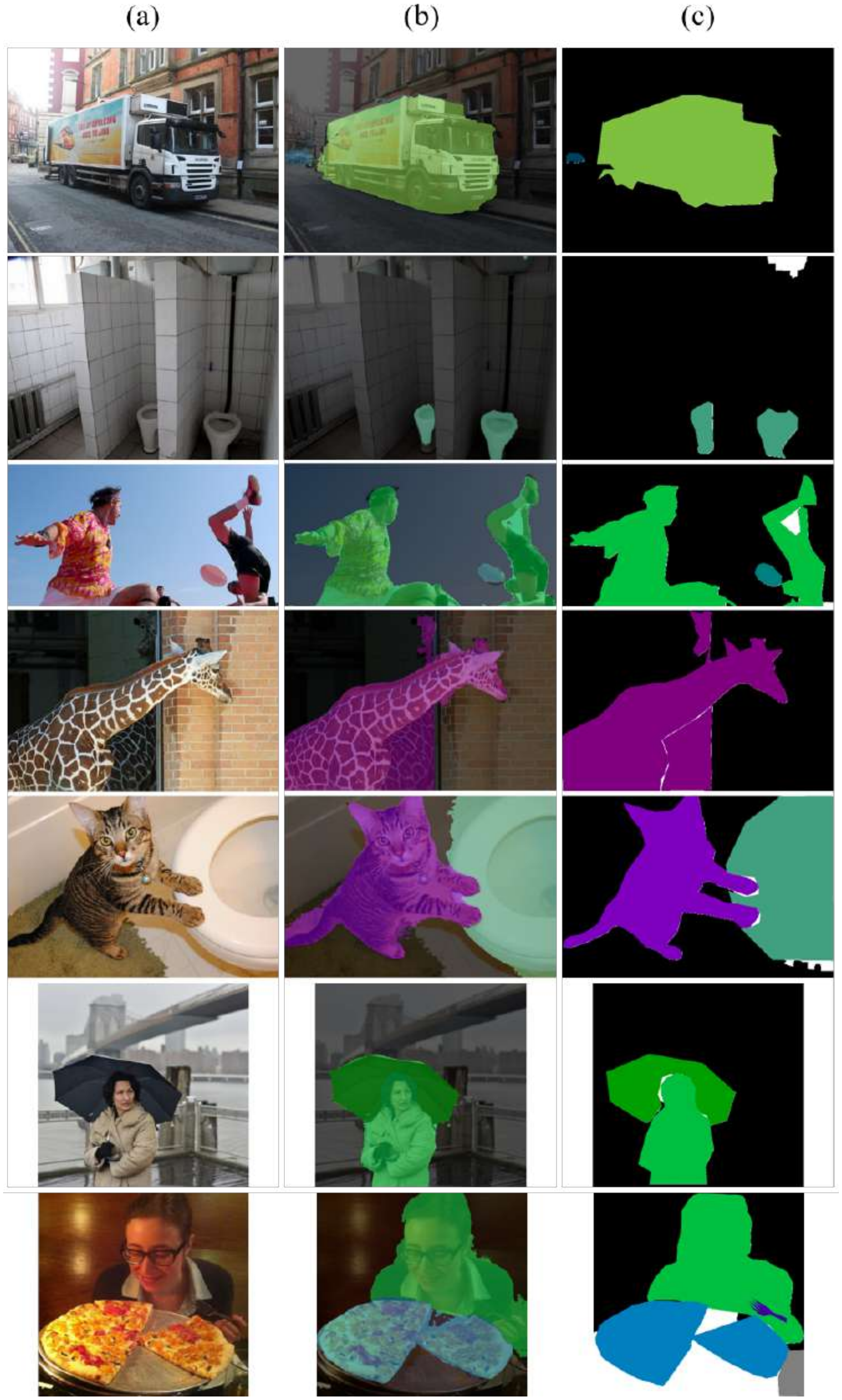

Figure A9. Semantic segmentation results on MS COCO *validation* set. From (a) to (c): images, our Deeplab, GTs.