CLIP is Also an Efficient Segmenter: A Text-Driven Approach for Weakly Supervised Semantic Segmentation
- Supplementary Material -

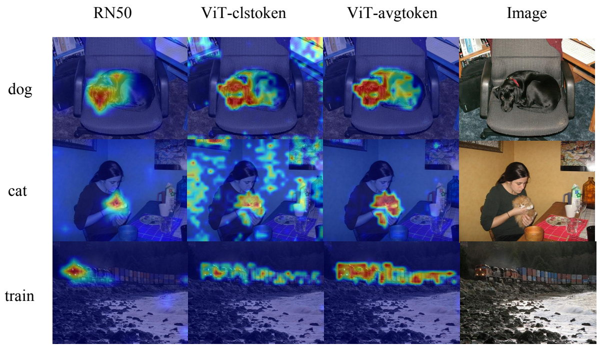

Figure S1. Qualitative comparisons between CNN and ViT architecture as well as clstoken and avgtoken for WSSS task.

A. More Analysis about GradCAM-CLIP

Pretrained CLIP models include two architectures, *i.e.*, ResNet-based and ViT-based. It is noteworthy that Grad-CAM is not only applicable to CNN-based architecture but also work on vision transformer. In our experiments, we find that the ResNet-based model suffers from the discriminative part domain problem heavily.In contrast, CAMs generated by ViT tend to cover more parts of objects. The qualitative and quantitative results can be found in Fig. S1 and Tab. S1, respectively. We adopt CLIP-pretrained ViT-B-16 in all our experiments.

Besides, ViT [4] tends to use an extra class token to get classification logits and compute the loss. An alternative is to perform average pooling on remaining tokens. The classification performances of the two methods tend to be similar in previous works. However, when applying Grad-CAM to CLIP, we find that CAMs generated by these two methods are somewhat different. The latter method can localize objects more completely and accurately, as is shown in Fig. S1. We suppose that the classification task is image-level, yet localization is pixel-level or region-level. The clstoken contains semantic information of the whole image and focuses on the patches that contribute more to it, while the average value of remaining tokens could treat each token equally. The latter is more suitable for dense prediction tasks, especially for the multi-label setting. Results in Tab. S1 demonstrate the superiority of the average pooling token for the WSSS task. Furthermore, the *sharpness* of avgtoken is significantly smaller than clstoken. It implies that avgtoken can attend to more classes rather than make one class prominent. The results verify the rationality of our proposed metric as well.

Model	Initial	CAA refined	Shaprness
RN50	38.2	-	0.019
ViT-clstoken	43.8	62.4	0.021
ViT-avgtoken	58.6	70.8	0.004

Table S1. Quantitative comparisons between CNN and ViT architecture as well as clstoken and avgtoken for WSSS task.

Category	Sentence-level	Feature-level	CAM-level
bird	76.7	76.7	76.6
chair	48.4	48.4	47.7
person	63.2	63.8	65.8
tvminotor	57.2	57.2	53.9
all classes	70.8	70.8	70.6

Table S2. Comparison of different synonym fusion strategies on PASCAL VOC 2012 train set.

B. Comparisons of Different Synonym Fusion Strategies

We can perform synonym fusion in different stages. Without loss of generality, we divide it into three types: 1) sentence-level (before inputting into text-encoder), 2) feature-level (after text-encoder), 3) CAM-level (after CAM generation). We perform synonym fusion on 4 categories and compare the three strategies in Tab. S2. The results remain similar and merely varied slightly among these approaches for each category as well as all categories. Since the last two methods require multiple encode processes for each synonym, we adopt the time-efficient sentence-level fusion strategy in our experiments.

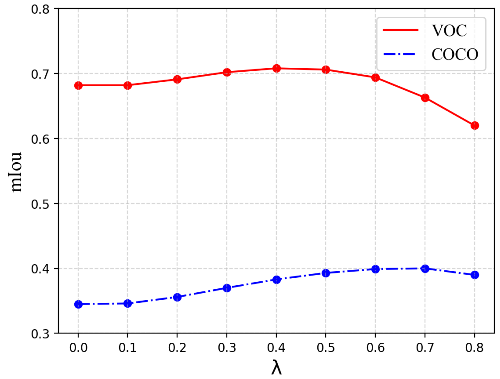

Figure S2. Effect of λ for the quality of generated CAMs on PASCAL VOC 2012 and part of COCO 2014 train set.

Confidence	[0.5, 0.8]	[0.8, 0.95]	[0.95, 1.0]
Frequency(%)	0.78	1.43	97.75
μ	0.7	0.8	0.95
mIoU	73.7	73.6	73.8

Table S3. The distribution of confidence and mIoU of final segmentation with different μ on VOC 12.

C. Hyper-parameter Selection for λ

In CAA module, we generate a class-aware mask for MHSA in the transformer. A parameter λ is used to binarize the CAM and generate some bounding boxes. In this part, we investigate the effect of λ on PASCAL VOC 2012 and COCO 2014 train set. Since the amount of COCO train set is tremendous, we only select the first 2000 images for research. We vary the threshold from 0 to 0.8 with an interval of 0.1. The results in Fig. S2 indicate that the best threshold varies on different datasets. We suppose that COCO is more complex and contains more objects in an image than PASCAL VOC on average. Therefore, a stricter threshold is required to identify regions belonging to the target class. In our experiment, we set λ to 0.4 and 0.7 for VOC and COCO, respectively.

D. Hyper-parameter Selection for μ in CGL

In the experiments, we found most pixels are confident enough after dense CRF postprocessing [5]. We calculate the confidence distribution on VOC (VOC's original ignored percentage is about 5.4%). Results in Tab. S3 indicate that only a small minority of pixels (mainly near object boundaries) have confidence lower than 0.95, and μ doesn't affect the segmentation performance remarkably. Therefore, we set μ to 0.95 in our experiments.

E. Training Details of DeepLabV2

For VOC, images are randomly scaled to [0.5, 0.75, 1.0, 1.25, 1.5] and cropped to 321x321. The batch size is set to 10, and iteration is 20k as default. For COCO, we use strong augment following [7]. Images are randomly scaled to [0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0] and 481x481 are cropped. The batch size and the number of training iterations are set to 5 and 100k, respectively. The initial learning rate is 2e-4 for imagenet-pretrained model and 2.5e-5 for COCO-pretrained model, with the polynomial learning rate decay $lr_{iter} = lr_{init}(1 - \frac{iter}{maxiter})^{\gamma}$, where $\gamma = 0.9$. We set $\mu = 0.95$ to ignore unconfident pseudo labels. Balanced cross-entropy loss is adopted for COCO training as in [6,7]. For testing, we adopt a multi-scale strategy and dense CRF to post-process with default hyper-parameters in [3].

F. Detailed Setting of Time and Memory Efficiency

We compare our proposed framework with classical AdvCAM [6], another language-supervised work CLIMS [8] and ViT-based work MCTFormer [9] in term of time and memory. All the experiments are conducted on a TITAN RTX GPU with 24 GB memory. We use their open-source code and follow the default procedure. When applying dense CRF, 20 num-workers are adopted for multiprocessing. The maximum memory occurs during the affinity network training stage, which is about 18GB for both PSA [2] and IRN [1]. With only 2GB memory, our training-free method could generate pseudo masks for PASCAL VOC 2012 train aug set (with 10582 images) within 1 hour. Note that adopting multiple GPUs or multiprocessing can further speed up this process.

G. Background Set

We define 25 class-related background categories for VOC, including $\{$*ground, land, grass, tree, building, wall, sky, lake, water, river, sea, railway, railroad, keyboard, helmet, cloud, house, mountain, ocean, road, rock, street, valley, bridge, sign*$\}$. For COCO, we simply remove $\{$*sign, keyboard*$\}$ since these categories have been defined in COCO categories.

H. More Qualitative Results

In Fig. S3, we provide more qualitative results of our generated pseudo labels and corresponding confidence maps on PASCAL VOC 2012 and MS COCO 2014 datasets. We can observe that our proposed framework produces satisfactory segmentation results. It is effective in both simple and complex scenes.

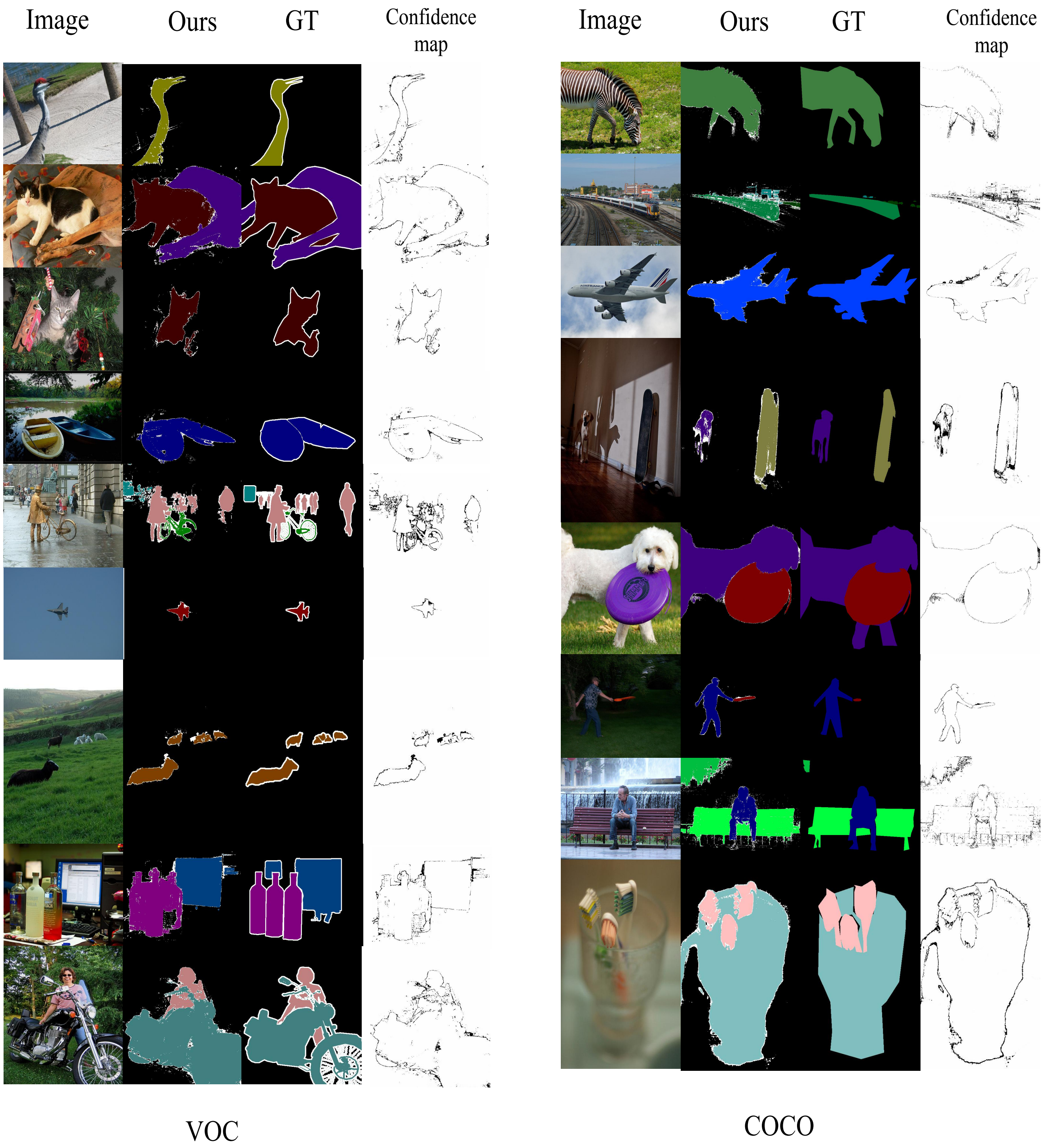

Figure S3. More visualizations on PASCAL VOC 2012 and MS COCO 2014 datasets.

References

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

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

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[5] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NeurIPS*, 2011.

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

[7] Minhyun Lee, Dongseob Kim, and Hyunjung Shim. Threshold matters in wsss: Manipulating the activation for the robust and accurate segmentation model against thresholds. In *CVPR*, 2022.

[8] Jinheng Xie, Xianxu Hou, Kai Ye, and Linlin Shen. CLIMS: Cross language image matching for weakly supervised semantic segmentation. In *CVPR*, June 2022.

[9] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *CVPR*, 2022.