Supplementary Material

A. More experiments with stronger supervision

In this section, we consider two additional scenarios when a stronger level of supervision is available; external semantic categories (see Appendix A.1), external training images and annotations (see Appendix A.2). Lastly, we present a comparison with (weakly) supervised baselines using the self-training technique (see Appendix A.3).

A.1. External segmentation categories

We investigate the effect of (a) hierarchical semantic categories and (b) external semantic categories from other sources.

Hierarchical semantic categories. Hierarchical semantic categories can be a stronger supervision for our artificial image creation. Specifically, we explore the semantic hierarchy as the language by supervising the model with multiple example words (*i.e.*, fine-grained categories) per single semantic category (*i.e.*, coarse categories). To this end, we first introduce the COCO Stuff benchmark [5] with the 27 coarse semantic categories, which remaps the original 171 fine-grained categories in the COCO stuff benchmark to the 27 coarse categories.[9] Then we augment each coarse category's words with those from its fine-grained categories for generating the artificial image; we slightly alter the artificial image creation in Sec. 2.3 to sample $h \cdot w$ coarse categories first, then perform additional sampling that actually assigns a word among the fine-grained categories associated with each coarse category. In our experiments, we empirically found that such hierarchical supervision significantly improves the performance of our method from 21.2 to 31.0 (+ 9.8) mIoU on the 27 coarse categories of the COCO Stuff benchmark. Furthermore, we provide a comparison with unsupervised semantic segmentation baselines on the coarse COCO Stuff benchmark. Tab. 6 summarizes the results; our method consistently and significantly outperforms all the existing baselines. For example, our method significantly outperforms STEGO [17] by achieving 31.0 mIoU in an image-free manner, while STEGO does 26.8, despite it requires task-specific images for training.

Model	Text Backbone	Image Backbone	Image Dataset	mIoU
IIC [19]	✗	ResNet-18	COCO (118k)	2.4
PiCIE + H. [9]	✗	ResNet-18	COCO (118k)	14.4
TransFGU [50]	✗	ViT-S/8	COCO (118k)	17.5
STEGO [17]	✗	ViT-S/8	COCO (118k)	26.8
CLIP† [32, 53]	CLIP-ResNet	ResNet-101	✗	6.6
MaskCLIP† [53]	CLIP-ResNet	ResNet-101	✗	6.9
OFA† [53]	OFA-Base	ResNet-101	✗	2.2
IFSeg (ours)†	OFA-Base	ResNet-101	✗	**31.0**

Table 6. **Comparison with unsupervised semantic segmentation baselines** on the COCO Stuff benchmark. We report the mIoU metric evaluated on the 27 coarse semantic categories of the COCO Stuff benchmark. † denotes that our post-processing is applied.

External semantic categories from other sources. Here, we validate the effect of external semantic categories from other sources. To this end, we perform IFSeg using the 150 semantic categories of the ADE20K benchmark [52] and then evaluate it on the 15 unseen categories of the COCO Stuff benchmark.[10] Interestingly, even though only 4 semantic categories (*road*, *tree*, *grass*, and *river*) intersect between training and evaluation, our method still achieves a significant performance of 54.1 mIoU, which is close to 54.6 mIoU of ours in Tab. 1, which potentially indicates that our method with external categories from other sources could learn transferable representations to novel semantic categories.

A.2. External training images and annotations

In this section, we investigate further improvements of ours when external training images and annotations as we described in Sec. 4.2. Overall, we empirically found that ours can achieve the best score compared to the baselines in both Tab. 2 and Tab. 3 when such stronger supervision is available; for example, ours in the last row of Tab. 7 shows the best score by fine-tuning task-specific images with corresponding pseudo-labels with 8k additional training iterations. Here, we generate

[9] The full list of hierarchy between the coarse and fine-grained categories are given in Tab. 15.

[10] We use the same vocabulary of the unseen semantic categories of the COCO Stuff in Sec. 4.1: *frisbee, skateboard, cardboard, carrot, scissors, suitcase, giraffe, cow, road, wall concrete, tree, grass, river, clouds, playing field*.

pseudo-labels via our pre-trained model following Zhou *et al.* [53]. Moreover, we also observed that fine-tuning ours in Tab. 2 with class-agnostic masks gives further enhancements from 17.4[11] to 20.9 mIoU with 60k additional training iterations following the configuration of ZSSeg [48], which is the strongest baseline and does 20.5 on the ADE20K benchmark (see Tab. 8). We note that our values using training images are reported without the post-processing, including Tab. 4.

Method	Backbone	Image Dataset	mIoU
IIC [19]	ResNet-18	COCO (118k)	0.6
PiCIE + H. [9]	ResNet-18	COCO (118k)	4.6
TransFGU [50]	ViT-S/8	COCO (118k)	11.9
MaskCLIP+ [53]	ResNet-101	COCO (118k)	18.0
CLIP† [32, 53]	ResNet-101	✗	4.5
MaskCLIP† [53]	ResNet-101	✗	13.7
OFA† [42]	ResNet-101	✗	1.5
IFSeg (ours)†	ResNet-101	✗	16.9
IFSeg (ours)	ResNet-101	COCO (118k)	**18.4**

Table 7. **Ablation study on the effect of external training images.** All models are evaluated on the 171 semantic categories of the COCO Stuff unsupervised segmentation benchmark. The last row indicates that fine-tuned result on training images of the COCO Stuff benchmark and corresponding pseudo labels generated by ours with 8k iterations. † denotes that our post-processing is applied.

Method	Text Backbone	Image Backbone	Image Dataset	Segmentation Label	mIoU
LSeg+ [15, 25]	ALIGN-BERT-Large [20]	ResNet-101	COCO (118k)	✓	13.0
OpenSeg [15]	ALIGN-BERT-Large [20]	ResNet-101	COCO (118k)	✓	15.3
ZSSeg [48]	CLIP-ViT-B [32]	ResNet-101	COCO (118k)	✓	20.5
CLIP† [32, 53]	CLIP-ResNet [32]	ResNet-101	✗	✗	3.9
MaskCLIP† [53]	CLIP-ResNet [32]	ResNet-101	✗	✗	11.3
OFA† [42]	OFA-Base [42]	ResNet-101	✗	✗	0.5
IFSeg (ours)†	OFA-Base [42]	ResNet-101	✗	✗	16.8
IFSeg (ours)	OFA-Base [42]	ResNet-101	COCO (118k)	✓	**20.9**

Table 8. **Ablation study on the effect of external segmentation annotations.** All models are evaluated on the 150 semantic categories of the ADE20K benchmark. The last row indicates that our fine-tuned result on training images of the COCO Stuff benchmark and corresponding class-agnostic segmentation masks with 60k iterations following the configuration of ZSSeg. † denotes that our post-processing is applied.

A.3. Comparison on weakly-supervised zero-shot transfer scenario

In this section, we present a comparison between our method and (weakly) supervised baselines using the self-training technique [4], which has been widely used in VL-driven zero-shot segmentation literature. Inspired by the weakly-supervised zero-shot transfer recipe proposed in MaskCLIP+ [53], we consider a weakly-supervised variant of our model, named IFSeg+, which is trained based on the ground truth segmentation labels for the 156 seen classes, the pseudo labels for the 15 unseen classes produced by the pre-trained IFSeg,[12] and an additional set of pseudo labels that are produced by IFSeg+ model itself during training. To be specific, we train IFSeg+ using the pre-trained OFA-Base [42] checkpoint, leveraging the ground truth segmentation labels (for the seen 156 classes) and the pseudo labels (for the unseen 15 classes) generated by the pre-trained IFSeg during initial 15k training iterations. Subsequently, we replace the pseudo labels generated by IFSeg with those generated by the IFSeg+ itself. We then apply the self-training technique [4] for the remaining 66k training iterations.

For evaluation, we follow the protocol of COCO Stuff seen → unseen zero-shot transfer scenario considered by prior works [4, 7, 16, 31, 45, 48, 53] where all 171 semantic categories of the COCO Stuff have to be predicted, then the mIoU metrics for the seen and the unseen categories are individually considered (*i.e.*, mIoU(U) and mIoU(S)), as well as their harmonic mean (*i.e.*, hIoU). Tab. 9 summarizes the results; our method (*i.e.*, IFSeg+) can achieve significant segmentation performances

[11] We empirically found that using hierarchical semantic categories for the ADE20K benchmark also improves the performance from 16.8 to 17.4 mIoU score. The hierarchy is publicly available at https://groups.csail.mit.edu/vision/datasets/ADE20K/.

[12] We use the pre-trained IFSeg checkpoint having 61.6 mIoU in Tab. 4.

compared to all the baselines. For example, IFSeg+ scored 2.1, 3.7, and 3.2 higher points than MaskCLIP+ [53] in terms of mIoU(U), mIoU(S), and hIoU, respectively. We note that our post-processing technique is not applied to the weakly-supervised zero-shot models, as the effect of the technique diminishes after using the real images and annotations during training as discussed in Sec. 2.3. For example, applying the post-processing ($K = 3$ with 25 iterations) even degrades the mIoU(U) scores of IFSeg+ and MaskCLIP+, dropping from 56.8 to 55.2, and from 54.7 to 54.5, respectively.

Method	Text Backbone	Image Backbone	Image Dataset	Segmentation Label	mIoU(U)	mIoU(S)	hIoU
ZS5 [4]	word2vec [29]	ResNet-101	COCO (118k)	✓(156 seen)	10.6	34.9	16.2
CaGNet [16]	word2vec [29], fasttext [21]	ResNet-101	COCO (118k)	✓(156 seen)	13.4	35.3	32.6
SIGN [7]	word2vec [29], fasttext [21]	ResNet-101	COCO (118k)	✓(156 seen)	15.2	36.4	21.4
SPNet [45]	word2vec [29], fasttext [21]	ResNet-101	COCO (118k)	✓(156 seen)	26.9	34.6	30.3
STRICT [31]	word2vec [29], fasttext [21]	ResNet-101	COCO (118k)	✓(156 seen)	30.3	35.3	32.6
ZSSeg [48]	ALIGN-BERT-Large [20]	ResNet-101	COCO (118k)	✓(156 seen)	43.6	39.6	41.5
MaskCLIP+ [53]	CLIP-ResNet [32]	ResNet-101	COCO (118k)	✓(156 seen)	54.7	38.2	45.0
IFSeg+ (ours)	OFA-Base [32]	ResNet-101	COCO (118k)	✓(156 seen)	**56.8**	**41.9**	**48.2**

Table 9. **Comparison with (weakly) supervised baselines under the seen→unseen transfer scenario.** We report the mIoU metric evaluated on the 15 unseen and the 156 seen semantic categories of the COCO Stuff benchmark and their harmonic mean, denoted by mIoU(U), mIoU(S), and hIoU, respectively. All models are trained on segmentation labels of the 156 seen categories (supervised training) and pseudo-labels of the 15 unseen categories (self-training), where "Image Dataset" denotes the dataset required for training.

B. Ablation study on hyperparameters

In this section, we perform an ablation study to understand the effect of hyperparameters of our method, namely the iteration count and K-nearest neighbors used in the post-processing, the sampling range S for the artificial image, and the use of cross-attention mechanism in our transformer decoder.

Post processing. We first examine the effect of the iteration count and the number of nearest neighbor K in our post-processing across an array of $\{0, 1, 10, 25, 50\}$ iteration count and $K \in \{2, 3, 5, 8\}$. As shown in Tab. 10, the effect of iteration counts becomes saturated after 25 iterations, and our method could be further improved with a larger K (*e.g.*, $K = 8$). We note that the evaluations are performed under the zero-shot semantic segmentation on the 15 unseen semantic categories of the COCO Stuff. We also note that 0 iteration is equivalent to not performing the post-processing.

Iteration	0	1	10	25	50
mIoU	46.8	51.2	54.9	55.6	55.5

(a) Varying iteration counts with $K = 3$.

K	2	3	5	8
mIoU	50.1	55.6	59.7	61.4

(b) Varying K with iteration counts of 25.

Table 10. **Ablation studies on varying the iteration count and the number of nearest neighbor K.** All models are trained and evaluated on the 15 unseen semantic categories of the COCO Stuff benchmark.

Method	PP	Backbone	Zero-shot (mIoU)	Cross-dataset (mIoU)	Unsupervised (mIoU)
CLIP [32, 53]	✓	ResNet-101	**12.3**	**3.7**	**4.6**
CLIP [32, 53]	✗	ResNet-101	11.6	3.6	4.4
MaskCLIP [53]	✓	ResNet-101	**24.8**	**10.3**	**12.7**
MaskCLIP [53]	✗	ResNet-101	23.7	8.8	10.8
IFSeg (ours)	✓	ResNet-101	**54.6**	**16.8**	**16.9**
IFSeg (ours)	✗	ResNet-101	47.0	14.0	14.3

Table 11. **Effects of the post-processing on varying image-free approaches.** We report the mIoU metric with and without the post-processing, evaluated on the zero-shot (the 15 unseen categories in COCO Stuff), the cross-dataset (COCO→ADE20K), and the unsupervised (all the 171 categories in COCO Stuff) semantic segmentation scenarios. "PP" denotes our post-processing is applied.

Next, we present the mIoU of the image-free models (*i.e.*, CLIP [32, 53], MaskCLIP [53], and IFSeg) without our post-processing evaluated on the zero-shot (the 15 unseen categories in COCO Stuff), the cross-dataset (COCO→ADE20K), and the unsupervised semantic segmentation (all the 171 categories in the COCO Stuff) scenarios in Tab. 11. Overall, our post-processing positively affects the mIoU of all baselines (*e.g.*, 23.7 mIoU → 24.8 mIoU for MaskCLIP on the zero-shot semantic segmentation scenario). Regardless of whether or not the post-processing is applied, however, IFSeg is always the best-performing image-free model in all the scenarios.

Artificial image. Here, we investigate the effect of varying sampling range k for our artificial image generation. Tab. 12 summarizes results; interestingly, optimal values of $k = 16$ and $K = 8$ (of the post-processing) give our significant further improvements from 55.6 to 66.0 (+ 10.4) mIoU score on the 15 unseen semantic categories of the COCO Stuff benchmark. We remark that the values of h and w in Eq. (17) are randomly sampled from $\{1, 2, ..., k\}$. Regarding this, the last row in Tab. 12 shows that removing randomness from sampling h and w harms overall improvements.

S	$(h, w) \sim \{1, 2, ..., S\}$	Post-processing with $K = 3$	Post-processing with $K = 8$
8	$(h, w) \sim \{1, 2, ..., 8\}$	55.8	64.3
16	$(h, w) \sim \{1, 2, ..., 16\}$	**57.8**	**66.0**
32	$(h, w) \sim \{1, 2, ..., 32\}$	55.6	61.4
32	$(h, w) = (32, 32)$	47.7	56.1

Table 12. **Ablation studies on varying sampling range S for our artificial image generation.** We also report two different nearest neighbor hyperparameters $K \in \{3, 8\}$ of the post-processing. The last row reports the deterministic setup of $(h, w) = (32, 32)$ for generating our artificial images. The reported values are mIoU scores on the 15 unseen semantic categories of the COCO Stuff benchmark.

On the other hand, one may consider the recent VL prompt learning method [54, 55] as an option for efficiently adapting a VL model to the semantic segmentation task. However, we would like to emphasize that our primary interest lies in image-free scenarios. Simply plugging the prompt learning into the image-free setting is non-trivial, as prompting cannot replace the training images and labels required to learn the segmentation task. Nonetheless, formulating image-free semantic segmentation within the context of the prompt learning framework could be an interesting direction for future research.

The cross-attention mechanism. We validate the effect of the cross-attention mechanism in our transformer decoder. To this end, we train our model on the zero-shot (the 15 unseen categories in COCO Stuff) semantic segmentation scenario without providing the contextualized embedding (Eq. (7)) for the cross-attention mechanism. As a result, we observed a significant degradation in segmentation performance, dropping from 55.6 → 22.6 mIoU after removing the cross-attention. We note that the use of cross-attention is a default setting during the VL pre-training in our framework, and maintaining the cross-attention during fine-tuning would be beneficial for stability.

C. Image-free baselines with ViT backbone

In this section, we present the mIoU of the image-free baselines, CLIP [32, 53] and MaskCLIP [53], with the stronger ViT-B/16 image backbones evaluated on the zero-shot (the 15 unseen categories in COCO Stuff) semantic segmentation scenario in Tab. 13. Overall, ViT-B/16 brings performance improvements thanks to its advanced visual representation compared to the ResNet-101 backbone. Nonetheless, the performance of our IFSeg is superior to these baselines even if it uses the ResNet-101 as the image backbone model, unchanged from the trends observed in Tab. 1.

Method	Text Backbone	Image Backbone	mIoU
CLIP† [32, 53]	CLIP-ViT-B/16 [32]	ViT-B/16	12.9
MaskCLIP† [53]	CLIP-ViT-B/16 [32]	ViT-B/16	37.0
IFSeg (ours)†	OFA-Base [42]	ResNet-101	**55.6**

Table 13. **Comparison with image-free baselines under the zero-shot semantic segmentation (the 15 unseen categories in COCO Stuff) scenario.** We report the mIoU metric evaluated on the 15 unseen semantic categories of the COCO Stuff benchmark. † indicates models with our post-processing applied.

D. Compatibility analysis

We here validate the compatibility of our method with another encoder-decoder VL model, CLIPCap [30]. Note that CLIPCap is a fine-tuned CLIP-ViT-B/32 model for an image-to-text captioning task on the Conceptual Captions benchmark [36].

Specifically, CLIPCap utilizes GPT2 [33] as a text generator, and we also do it as the segmentation decoder in our framework.

In order to create our artificial image under CLIP's dual-encoder design, we utilize CLIP text encoder's sentence-level feature as the word embedding for semantic categories, directly following the prompt engineering procedure by MaskCLIP [53]. For example, an artificial image patch for a *dog* category is an ensemble of prompts like "*a photo of the dog*" and "*a painting of a dog*."[13] Then, similar to ours incorporated with the OFA framework, we fine-tune the text generator of ClipCap to predict semantic segmentation of the artificial image and evaluate the performance on the 15 unseen semantic categories of the COCO Stuff benchmark. We note that, in order to deal with the prefix-based design of ClipCap (*i.e.*, a single token in the CLIP representation space is mapped to multiple tokens in the text generator space), we decode each token individually.

Tab. 14 summarizes the compatibility experiments; our method is well-incorporated with CLIPCap and even significantly outperforms CLIP and MaskCLIP [53] baselines, which also have the same CLIP backbone. For example, our method achieves the best mIoU score of 25.8 on the 15 unseen semantic categories of the COCO Stuff benchmark compared to the baselines having the same CLIP backbone. These results demonstrate the broad applicability of our method with various pre-trained VL models and lead them to perform semantic segmentation in an image-free manner.

Method	Pretrain	Image Backbone	Text Deocder	mIoU
CLIP† [32, 53]	CLIP [32]	CLIP-ViT-B/32	✗	4.8
MaskCLIP† [53]	CLIP [32]	CLIP-ViT-B/32	✗	20.7
IFSeg (ours)	CLIPCap [30]	CLIP-ViT-B/32	GPT2 [33]	**25.8**

Table 14. **Ablation study on compatibility with other encoder-decoder VL models.** We denote that our image-free approach is applied to CLIPCap, which is an image captioning model built upon pre-trained CLIP. All models are evaluated on the 15 unseen semantic categories of the COCO Stuff benchmark. † denotes that our post-processing is applied.

E. Implementation details

Image Pre-processing. We preprocess images using the official codebase[14] of OFA [42] framework and `mmsegmentation`[15]. Specifically, we normalize the image with the mean and standard deviation values of 0.5. We also resize the short sides of images keeping the aspect ratio. For all experiments, we resize the short sides to 512, in order to ensure a fair comparison with the strongest baselines MaskCLIP+ [53] and DenseCLIP [34].

Text Pre-processing. We generate prompt text following the "*task description* + *category enumeration*" protocol of the VQA task [42]. Precisely, we use "*what is the segmentation of the image?*" as the task description, and "*object: category1 category2 ... categoryN*" as the category enumeration. For the tokenization and embedding, we directly incorporate the pre-trained BPE tokenizer and embedding matrix provided by the codebase of OFA [42] framework.

Evaluation Details. For a fair comparison, we perform the *whole inference* evaluation protocol (*i.e.*, predicting the rectangular-shaped output at once) for image-free based approaches (*e.g.*, Tabs. 1 to 4) following their strongest baseline, MaskCLIP+ [53], and the *sliding inference* evaluation protocol (*i.e.*, concatenating square-shaped crops of the original rectangular-shaped image) for supervised approaches (*e.g.*, Tab. 5) following their strongest baseline, DenseCLIP [34].

Visual Feature-based Post-processing. In our post-processing, we utilize features of the image backbone network (*i.e.*, ResNet) from the OFA [42] encoder. For a fair comparison, we also apply the post-processing for our re-implemented baselines of OFA [42], CLIP [32], and MaskCLIP [53] with their image backbone networks. For example, in the case of CLIP [32] and MaskCLIP [53]), we utilize the final patch-wise outputs of the ViT-B/16 image backbone as the post-processing features.

Viusalization Details. Exclusively for the visualizations of the image-free models (Figs. 1 and 4), we introduce additional post-processing with DenseCRF [24] and its third-party implementation[16]. Note that smoothing outputs with DenseCRF can provide qualitatively sharper segmentation results by clustering prediction outputs according to the edges of the raw RGB images. However, we remark that DenseCRF is *never* used for the reported values of experimental results for a fair comparison.

[13] We refer the readers to the codebase of MaskCLIP [53] for the full list of prompt templates; `https://github.com/chongzhou96/MaskCLIP`.

[14] `https://github.com/OFA-Sys/OFA`.

[15] `https://github.com/open-mmlab/mmsegmentation`.

[16] `https://github.com/lucasb-eyer/pydensecrf`.

Coarse category	Fine-grained category
animal	giraffe, zebra, bear, elephant, cow, sheep, horse, dog, cat, bird
sports	tennis racket, surfboard, skateboard, baseball glove, baseball bat, kite, sports ball, snowboard, skits, frisbee
accessory	suitcase, tie, handbag, eye glasses, shoe, umbrella, backpack, hat
outdoor	bench, parking meter, stop sign, street sign, fire hydrant, traffic light
vehicle	boat, truck, train, bus, airplane, motorcycle, car, bicycle
person	man, woman, child, boy, girl
indoor	hair brush, toothbrush, hair drier, teddy bear, scissors, vase, clock, book
appliance	blender, refrigerator, sink, toaster, oven, microwave
electronic	cell phone, keyboard, remote, mouse, laptop, tv
furniture (things)	door, toilet, desk, window, dining table, mirror, bed, potted plant, couch, chair
food (things)	cake, donut, pizza, hot dog, carrot, broccoli, orange, sandwich, apple, banana
kitchen	bowl, spoon, knife, fork, cup, wine glass, plate, bottle
water	waterdrops, sea, river, fog, lake, ocean
ground	playingfield, platform, railroad, pavement, road, gravel, mud, dirt, snow, sand
solid	hill, mountain, stone, rock, wood
sky	clouds
plant	straw, moss, branch, flower, leaves, bush, tree, grass
structural	railing, net, cage, fence
building	roof, tent, bridge, skyscrapper, house
food (stuff)	vegetable, salad, fruit
textile	banner, pillow, blanket, curtain, cloth, clothes, napkin, towel, mat, rug
furniture (stuff)	stairs, light, counter, mirror, cupboard, cabinet, shelf, table, desk, door
window	blind window
floor	stone floor, marble floor, wood floor, tile floor, carpet
ceiling	tile ceiling
wall	concrete wall, stone wall, brick wall, wood wall, panel wall, tile wall
raw material	metal, plastic, paper, cardboard

Table 15. **The full list of hierarchical semantic categories** of the COCO Stuff benchmark. Each coarse category is paired with given fine-grained categories, following the label hierarchy of Caesar *et al.* [5].

F. Additional qualitative results

In this section, we present visualizations of segmentation results obtained by baselines and our method in different evaluation settings. Specifically, we first consider the weakly-supervised scenario (zero-shot transfer) in Tab. 9 and compare the result between our IFSeg+ and MaskCLIP+, the strongest baseline in the scenario. Next, we also consider the fully-supervised semantic segmentation scenario in Tab. 5 and compare the result between the supervised IFSeg and DenseCLIP baseline.

F.1. Weakly-supervised zero-shot transfer scenario

The visualizations of segmentation results obtained by MaskCLIP+ and ours under the COCO Stuff seen→unseen zero-shot transfer scenario are present in Fig. 5. Following the protocol in Appendix A.3 we evaluate and visualize the 15 unseen classes of the COCO Stuff benchmark. Overall, it shows that our method can predict the segmentation that is more consistent with the groud-truth (GT) segmentation than the MaskCLIP+ baseline.

F.2. Fully-supervised semantic segmentation scenario

The visualizations of segmentation results obtained by DenseCLIP and ours under the ADE20k semantic segmentation benchmark are present in Fig. 6. We evaluate and visualize the total 150 classes of the ADE20k dataset. As depicted by the quantitative mIoU score in Tab. 5 and some visualization cases in Fig. 6, ours shows results that are more consistent with the groud-truth (GT) segmentation than the DenseCLIP baseline. However, we note that DenseCLIP and ours both tend to produce satisfactory prediction results for most samples since they are trained in a fully-supervised way using the ground-truth segmentation annotations.

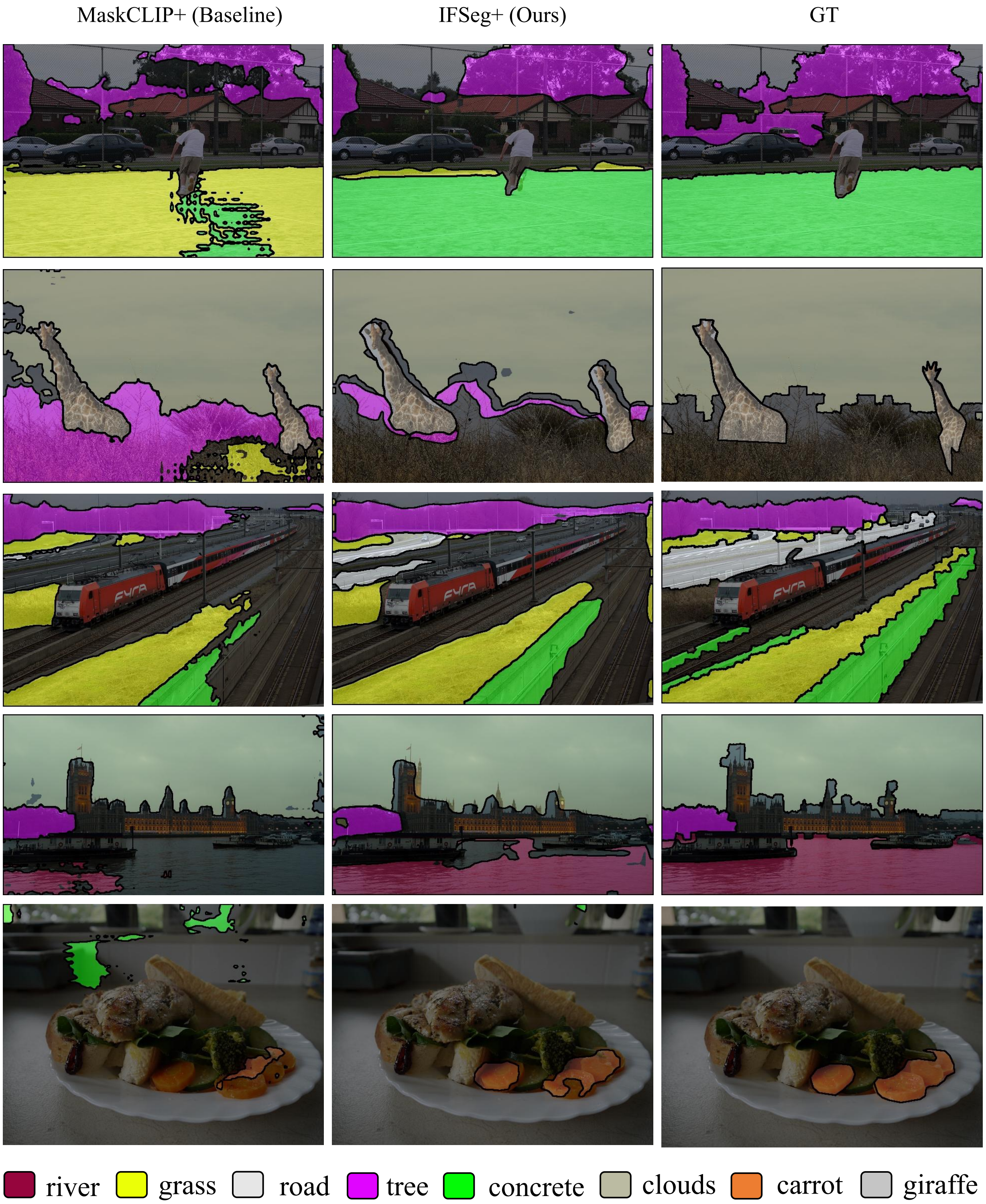

Figure 5. **Visualization of segmentation results under the weakly-supervised zero-shot transfer scenario.** We visualize the segmentation results of IFSeg+ (ours) and MaskCLIP+ (baseline). Qualitatively observed, IFSeg+ can predict the segmentation that is more consistent with the groud-truth (GT) segmentation than the MaskCLIP+ baseline. Best viewed in color.

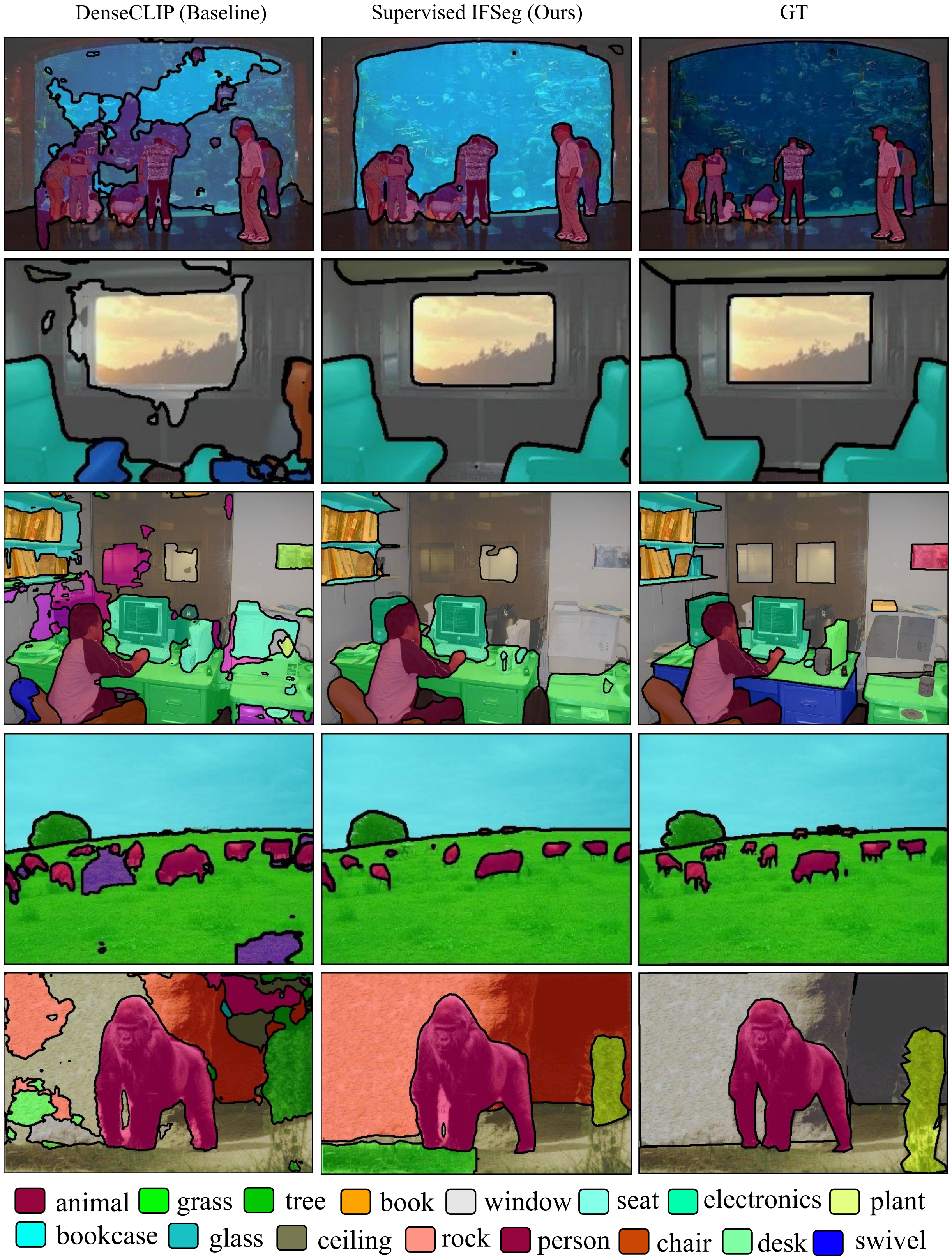

Figure 6. **Visualization of segmentation results under the fully-supervised semantic segmentation scenario.** We visualize the segmentation results of Supervised IFSeg (ours) and DenseCLIP (baseline). Although both the models are trained in a fully-supervised manner, our IFSeg tends to produce more accurate predictions than DenseCLIP. We utilize the class colors defined by the `mmsegmentation` in `https://github.com/open-mmlab/mmsegmentation/blob/master/mmseg/datasets/ade.py`. For clarity, we denote the 14 classes with the largest segmentation regions in this example. Best viewed in color.