# Supplementary Material
# Pixel Contrastive-Consistent Semi-Supervised Semantic Segmentation

Yuanyi Zhong[1], Bodi Yuan[2], Hong Wu[2], Zhiqiang Yuan[2], Jian Peng[1], Yu-Xiong Wang[1]

[1] University of Illinois at Urbana-Champaign
{yuanyiz2, jianpeng, yxw}@illinois.edu

[2] X, The Moonshot Factory
{bodiyuan, wuh, zyuan}@google.com

## A. Additional Implementation Details

In Tab. A.1, we list the DeepLab-related hyper-parameters. They take either the recommended default values by [1] or the values from the public code of [8]. The learning rate is linearly annealed from $0.007$ to $0$ as training proceeds. The weight decay is 1e-4 for ResNets [5] and 4e-5 for Xception [2]. The input train crop size is set to $513 \times 513$ for all datasets. The eval crop size is set to $513 \times 513$ for VOC [4], $641 \times 641$ for COCO [6], and $1,025 \times 2,049$ for Cityscapes [3]. The DeepLab encoder output stride is chosen as 16. In the VOC experiments, the model is trained for 30,000 gradient steps that are distributed on 4 asynchronous replica workers. Each worker has 2 GPUs. The Cityscapes and COCO experiments require longer training duration, which takes a total of 130,000 and 200,000 steps, respectively, on 8 asynchronous workers of 2 GPUs. For both the labeled data and unlabeled data branches, the training batch size per GPU is 4, or equivalently 8 images per worker.

Tab. A.1: Summary of hyper-parameter settings. Hyper-parameters take either the recommended default values by [1] or the values from the public code of [8].

| Hyper-parameter | VOC | Cityscapes | COCO |
|---|---|---|---|
| Network | ResNet50/101 | ResNet50/101 | Xception65 |
| Weight decay | 1e-4 | 1e-4 | 4e-5 |
| Train crop size | $513 \times 513$ | $513 \times 513$ | $513 \times 513$ |
| Eval crop size | $513 \times 513$ | $1,025 \times 2,049$ | $641 \times 641$ |
| Output stride | 16 | 16 | 16 |
| Atrous rates | [6, 12, 18] | [6, 12, 18] | [6, 12, 18] |
| Train steps | 30,000 | 130,000 | 200,000 |
| Learning rate | $0.07 \rightarrow 0$ | $0.07 \rightarrow 0$ | $0.07 \rightarrow 0$ |
| Dist. workers | 4 | 8 | 8 |
| GPUs/worker | 2 V100(16G) | 2 V100(16G) | 2 V100(16G) |
| Batch size/GPU | 4 | 4 | 4 |

Tab. B.1: Ablation study on the dimension of the feature projection layer in the VOC 1/8 split and ResNet-50 setting.

| Dimension | 64 | 128 | 256 | 512 |
|---|---|---|---|---|
| VAL mIoU (%) | 64.12 | **64.63** | 64.07 | 63.74 |

Tab. B.2: Comparison between the supervised baseline, the label-consistent-only version (denoted as 'PC$^2$Seg w/ LC') and the joint label-consistent and feature-contrastive version of PC$^2$Seg with the VOC splits and ResNet-50 backbone.

| Method/Split | 1 (1464) | 1/2 (732) | 1/4 (366) | 1/8 (183) | 1/16 (92) |
|---|---|---|---|---|---|
| Supervised | 68.81 | 65.73 | 57.76 | 49.57 | 43.97 |
| PC$^2$Seg w/ LC (Ours) | 71.95 | 70.88 | 66.71 | 63.75 | 56.32 |
| PC$^2$Seg (Ours) | **72.26** | **70.90** | **67.62** | **64.63** | **56.90** |

Tab. B.3: Comparison between the supervised baseline, the label-consistent-only version (denoted as 'PC$^2$Seg w/ LC') and the joint label-consistent and feature-contrastive version of PC$^2$Seg with the Cityscapes 1/8 split and ResNet-50 backbone.

| Method | Supervised | PC$^2$Seg w/ LC (Ours) | PC$^2$Seg (Ours) |
|---|---|---|---|
| VAL_FINE mIoU (%) | 68.06 | 71.79 | 72.11 |

## B. Additional Ablation Results

**Dimension of Projection Layer.** The impact of the dimension of the projection layer in our PC$^2$Seg method is studied in Tab. B.1 under the VOC 1/8 ResNet-50 setting. For the results reported in the main paper, we chose the dimension as 128. We found that alternative values do not bring gains over the chosen value if other hyper-parameters are fixed. Alternative dimension values may require re-tuning some hyper-parameters to achieve better results.

**Label Consistent Only.** In Tab. 9 and Paragraph "Comparison to Other Label-Space and Feature-Space Losses" of Sec. 4.3 in the main paper, we have compared PC$^2$Seg

with its label-consistent-only version in the VOC 1/8 split setting. Here, we provide additional results with other data splits, which support the claim that the joint label-consistent and feature-contrastive regularization performs better than the label-consistent regularization alone. The label-consistent version essentially removes the pixel contrastive loss and keeps everything else unchanged. The VOC results are shown in Tab. B.2. We have a similar observation with the Cityscapes 1/8 split in Tab. B.3, where the label-consistent-only version achieves 71.79% validation mIoU in comparison to 72.11% mIoU of full PC$^2$Seg with the joint contrastive-consistent regularization.

## C. Training Time and Computational Cost

Since there is an additional unlabeled data branch in our semi-supervised method, the training inevitably takes longer time than the purely supervised baseline. As stated in the main paper, we measure the training time in the VOC ResNet-50 experiments. Our implementation of the supervised baseline took 38 minutes, while our PC$^2$Seg with label consistency and feature contrastive learning took roughly 80 minutes, and the label-consistent-only version of PC$^2$Seg took around 75 to 80 minutes. Such training time is comparable to existing approaches *within the semi-supervised setting – e.g.*, state-of-the-art PseudoSeg [8] also took about 80 minutes.

We further show the cost reduced by our negative sampling strategy through a simple calculation. The feature tensor shape in practice is $[4, 33, 33, 128]$ for a batch of 4 images (512-by-512 pixels). If using all pixels as negatives, we need to compute a $[4 \cdot 33^2, 4 \cdot 33^2]$ inner-product matrix ($19M$ elements) with $(4 \cdot 33^2)^2 \cdot 128 = 2,428M$ MulAdd operations. But, if we only draw 200 negative pixels, it is reduced to computing a $[4 \cdot 33^2, 200]$ matrix ($0.8M$ elements) with $4 \cdot 33^2 \cdot 200 \cdot 128 = 111M$ operations. Considering the negative sampling itself requires $(4 \cdot 33^2)^2 \cdot 20 = 379M$ operations to compare the pseudo labels (20 is the number of VOC classes), the overall floating point operations are about 5 times fewer.

## D. Visualization

To have a better understanding of PC$^2$Seg, we use two complementary approaches to visualize the results: (1) the t-SNE plot [7] of the feature spaces in Fig. D.1, and (2) the predicted segmentation masks in Fig. D.2 and Fig. D.3.

From the t-SNE plot, we observe that both the label-consistent-only and the joint contrastive-consistent variants of our method generate feature spaces that are more separable than the supervised baseline. In the decoder feature space, joint contrastive-consistent variant seems to increase slightly the margins between a few categories, compared with the label-consistent-only variant. From the predicted masks, we can see some success and failure cases of PC$^2$Seg. Please refer to the figure captions for detailed descriptions.

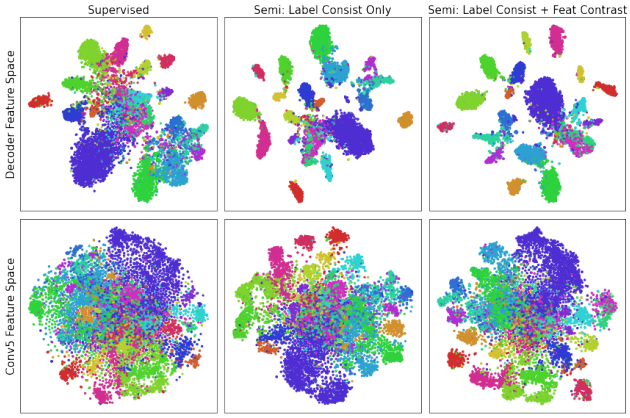


Fig. D.1: t-SNE [7] visualization of the Decoder and Conv5 (of ResNet-50) feature spaces generated by different methods on the VOC VAL images. The train set is the VOC 1/8 split. Conv5 is the feature layer where the pixel contrastive loss is applied. We randomly sample 10,000 data points to produce the t-SNE plot, and the perplexity parameter is set to 40. We observe clearer separation of semantic classes in the feature spaces with semi-supervised methods than that with the supervised baseline.

## References


[1] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 1

[2] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, 2017. 1

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 1

[7] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *JMLR*, 9(11):2579–2605, 2008. 2

[8] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. PseudoSeg: Designing pseudo labels for semantic segmentation. In *ICLR*, 2021. 1, 2

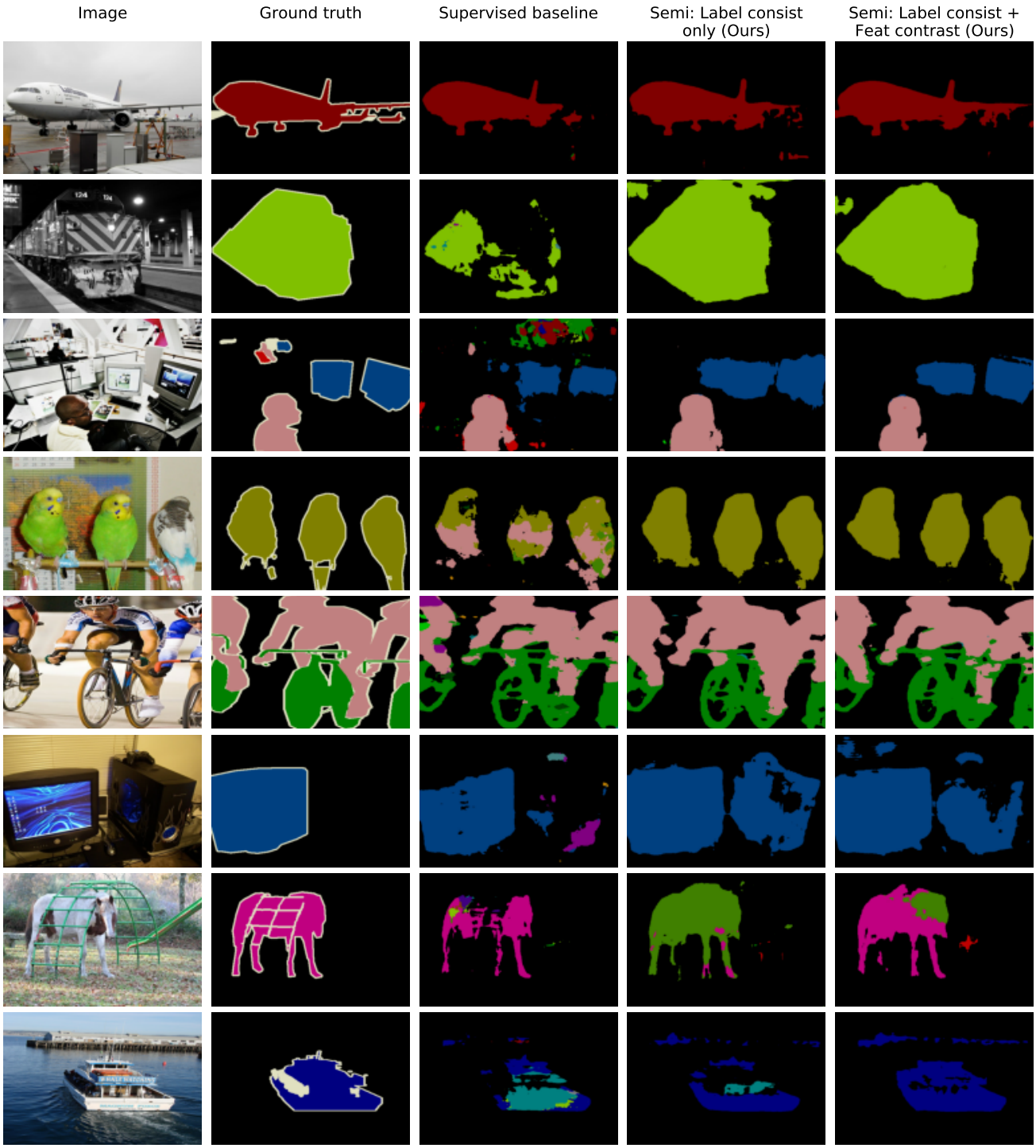


Fig. D.2: VOC 2012 prediction results. Models are trained in the 1/8 split setting. Images are from the VAL set. The white pixels in the ground-truth masks indicate ignored regions. *Overall, both the label-consistent-only and the contrastive-consistent variants of our semi-supervised method produce significantly less noisy predictions than the supervised baseline*. The consistency regularization is able to suppress some background artifacts as in the 3rd row example. Some other success cases include the airplane in the 1st row and the train in the 2nd row, where our final method covers fuller extents of the objects. Some failure cases include the monitor in the 3rd to last row, where the model mistakenly classifies the (possibly co-occurring) machine into the monitor class, the mis-classification of the horse in the 2nd to last row, and the false positives in the last row.

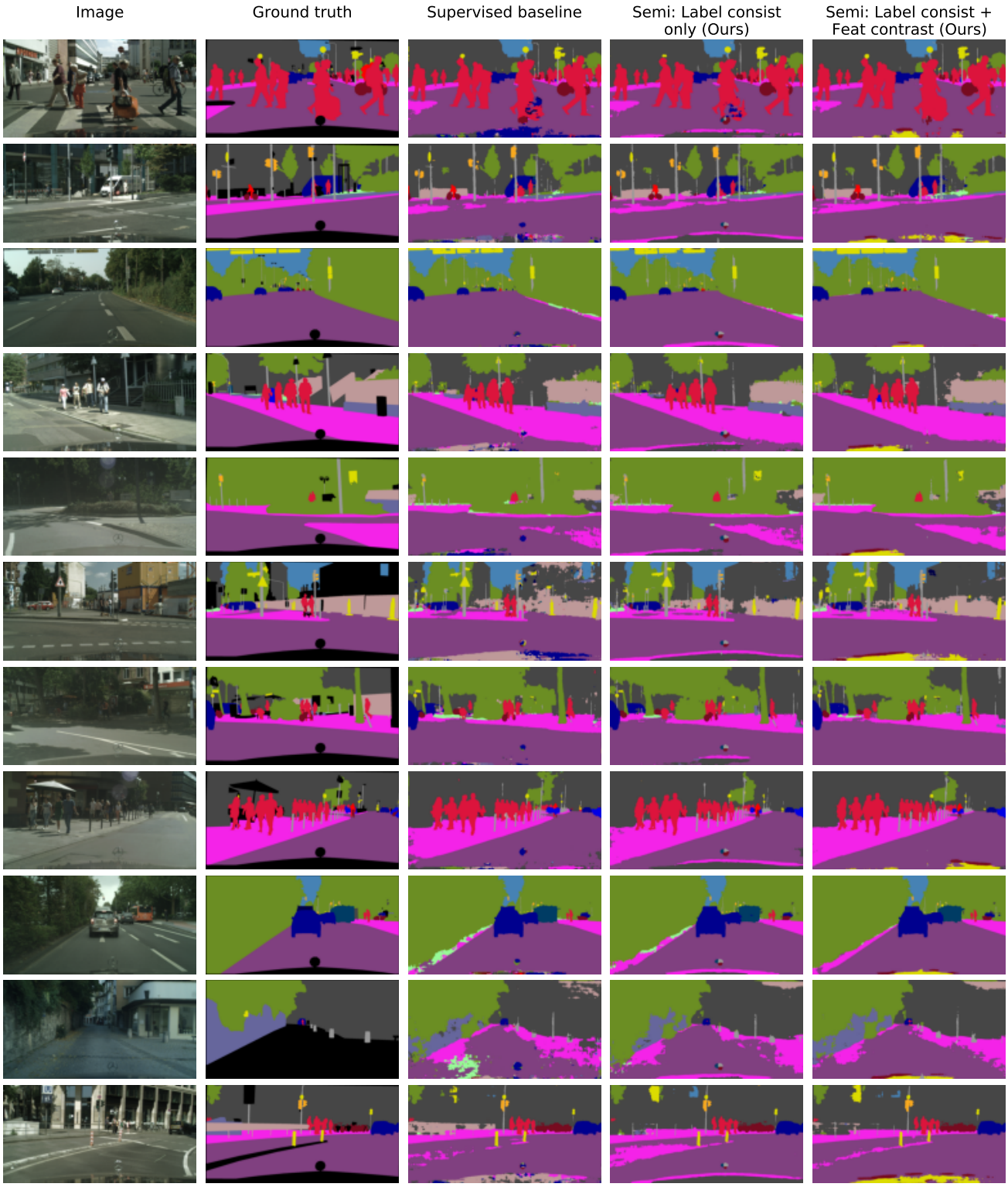


Fig. D.3: Cityscapes prediction results. Models are trained in the 1/8 split setting. Images are from the VAL_FINE set. The dark areas in the ground-truth masks indicate ignored regions. We notice that there exist some *substantially higher-quality* cases of our contrastive-consistent learning based semi-supervised method over others, such as the person's bag in the 1st row, the sidewalk in the 2nd row, and the upper traffic sign in the 3rd row.