Prototypical Pseudo Label Denoising and Target Structure Learning for Domain Adaptive Semantic Segmentation (Supplementary Material)

Pan Zhang[1] *, Bo Zhang[2], Ting Zhang[2], Dong Chen[2], Yong Wang[1], Fang Wen[2]
[1]University of Science and Technology of China [2]Microsoft Research Asia

1. Influences of Design Choices

Prototype initialization strategy. In our implementation, the proposed ProDA initializes the prototypes of the target domain according to the pseudo predictions for the target images. Alternatively, the target prototypes can also be initialized according to the ground truth labeling in the source domain. However, both choices have their pros and cons: the former suffers from the noises in the pseudo labels whereas the latter suffers from domain gap as the prototypes of the two domains may not accurately align. Table 1 shows that the two initialization strategies induce comparable results, as the prototypes are online updated and can rapidly converge to the true cluster centroids. The quantitative performance is measured on the dataset GTA5 $\rightarrow$ Cityscapes, whereas the other dataset shows similar results.

	source ground truth	target pseudo label
mIoU	53.6	53.7

Table 1: The performance of different target prototype initialization strategies. Here we only report the performance for the 1st training stage in the gta5 $\rightarrow$ Cityscapes task.

Strong augmentation. In the target structure learning, we take weak and strong augmentation views for the target image. We employ random crop for weak augmentation and explore the effects of different augmentation types for the strong augmented view. As shown in Table 2, random crop only gives the mIoU score 52.7, whereas adding RandAugment [1] and CutOut [2] respectively improve the mIoU by 0.78 and 0.5. The strongest augmentation gives the best performance, indicating the importance of data augmentation when learning the compact feature space for the target domain.

	crop	crop & RandAug	crop & Cutout	crop & RandAug & Cutout
mIoU	52.7	53.5	53.2	53.7

Table 2: The influence of various strong augmentations. Here we only report the performance for the 1st training stage in the gta5 $\rightarrow$ Cityscapes task.

Effect of temperature during prototypical denoising. We rely on the prototypical context to rectify the pseudo labels. We compute the softmax over feature distance to all the prototypes, and the softmax temperature τ influences the denoising effect and requires balancing: when $\tau \rightarrow 0$, only the nearest prototype dominates whereas $\tau \rightarrow \infty$ causes that all the prototypes are accounted equally. The influence of the temperature is shown in Table 3. We empirically set $\tau = 1$ in our experiments.

Symmetric cross-entropy loss. We employ the symmetric cross-entropy loss (SCE) for robust learning to stabilize the early training phase. The SCE has coefficients α and β that balance the cross-entropy and the reverse cross-entropy. Table 4 shows that the final result is not sensitive to these hyper-parameters if β is not too small. Here, we follow the suggested setting as [8], *i.e.*, $\alpha = 0.1, \beta = 1$.

The effect of loss weight. Table 5 shows that the final result is not sensitive to the KL loss weight (γ_1) and the regularization loss weight (γ_2). In GTA5 $\rightarrow$ Cityscapes, we set $\gamma_1 = 10$ and $\gamma_1 = 0.1$, while in SYNTHIA $\rightarrow$ Cityscapes, we set $\gamma_1 = 10$ and $\gamma_1 = 0$.

*This work is done during the first author's internship at Microsoft Research Asia.

	0.1	0.5	1	2	3	5	10
mIoU	48.8	52.1	53.7	51.9	47.5	44.9	40.9

Table 3: The effects of temperature during the prototypical denoising. Here we only report the performance for the 1st training stage in the gta5 $\rightarrow$ Cityscapes task.

α \ β	0.1	0.5	1	5
0.01	46.4	52.7	53.8	53.6
0.1	47.6	52.9	53.7	53.5
0.5	50.4	53.1	53.3	53.5
1	51.1	52.7	53.1	53.6

Table 4: The influence of α and β in the symmetric cross-entropy (SCE) loss. Here we only report the performance for the 1st training stage in the gta5 $\rightarrow$ Cityscapes task.

γ_1 \ γ_2	0.02	0.1	0.2
2	52.9	53.7	53.5
10	53.2	53.7	53.4
20	53.4	53.6	52.1
50	53.6	52.0	52.1

Table 5: The influence of the KL loss weight (γ_1) and the regularization loss weight (γ_2). Here we only report the performance for the 1st training stage in the gta5 $\rightarrow$ Cityscapes task.

2. Algorithm

The training procedure of our ProDA is summarized in Algorithm 1, which is composed of three stages. The first stage consists of prototypical pseudo label denoising and target structure learning. In the second and third stages, we apply knowledge distillation to a self-supervised model. For detailed equations and loss functions, please refer to our main paper.

Algorithm 1: ProDA

Input: training dataset: $(\mathcal{X}_s, \mathcal{Y}_s, \mathcal{X}_t)$; prototype momentum: λ; weak, strong augmentations: $\mathcal{T}$, $\mathcal{T}'$; the pretrained SimCLRv2 model: $h_\theta^{'}$; pseudo label selection threshold: T;

Output: the output model h_θ.

Warmup: $h_\theta = g_\theta \circ f_\theta \leftarrow (\mathcal{X}_s, \mathcal{Y}_s, \mathcal{X}_t)$ according to [5];
Generate soft pseudo label: $p_{t,0} \leftarrow h_\theta(\mathcal{X}_t)$;
Prototype initialization: $\eta_c \leftarrow (f_\theta, \mathcal{X}_t)$;
EMA model initialization: $\tilde{h}_\theta \leftarrow h_\theta$;
for $m \leftarrow 0$ **to** *epochs* **do**
    for $i \leftarrow 0$ **to** *len(X_t)* **do**
        Get source images $x_s^{(i)}$;
        Train the model h_θ using loss ℓ_{ce}^s;

        Get target images $x_t^{(i)}$;
        Calculate the denoising weight $\omega_t^{(i,k)}$;
        Update the pseudo label $\hat{y}_t^{(i,k)}$;
        Train model h_θ using loss ℓ_{sce}^t;

        Calculate the soft label $z_{\mathcal{T}}, z_{\mathcal{T}'}$;
        Train the model h_θ using loss ℓ_{kl}^t and ℓ_{reg}^t;

        Calculate the batch prototype η_c' ;
        $\eta_c \leftarrow \lambda\eta_c + (1-\lambda)\eta_c'$;
        Update the EMA model $\tilde{h}_\theta$;

for *stage* $\leftarrow 1$ **to** 2 **do**
    Generate the pseudo label: $\hat{y}_t \leftarrow \xi(h_\theta(\mathcal{X}_t), T)$;
    Student model initialization: $h_\theta^\dagger \leftarrow h_\theta^{'}$;
    for $m \leftarrow 0$ **to** *epochs* **do**
        for $i \leftarrow 0$ **to** *len(X_t)* **do**
            Get source images $x_s^{(i)}$;
            Tune the model $h_\theta^\dagger$ using loss ℓ_{ce}^s;

            Get target images $x_t^{(i)}$;
            Calculate the teacher probability $h_\theta(x_t^{(i)})$;
            Calculate the student probability $h_\theta^\dagger(x_t^{(i)})$;
            Tune the model $h_\theta^\dagger$ using loss ℓ_{ce}^t and KL loss;
    $h_\theta \leftarrow h_\theta^\dagger$;

3. Detailed Ablation study

Here we show a detailed ablation study for all the 19 classes on GTA5 → Cityscapes.

	components				road	sideway	building	wall	fence	pole	light	sign	vege.	terrace	sky	person	rider	car	truck	bus	train	motor	bike	mIoU	gain
init.	source				75.8	16.8	77.2	12.5	21.0	25.5	30.1	20.1	81.3	24.6	70.3	53.8	26.4	49.9	17.2	25.9	6.5	25.3	36.0	36.6	
	warm up				86.7	34.2	79.3	26.6	21.6	38.4	33.7	15.8	82.1	31.0	73.2	60.4	21.0	82.3	23.2	32.0	2.9	24.1	20.9	41.6	+5.0
	ST	SCE	PD	SL																				mIoU	gain
stage 1	✓				87.0	39.7	77.5	31.5	25.7	41.5	38.7	20.3	84.6	38.2	74.1	63.7	21.7	86.0	29.0	37.5	0.3	34.9	26.2	45.2	+8.6
	✓	✓			87.7	36.8	78.2	30.9	24.8	41.5	40.0	23.2	83.0	35.3	72.9	64.1	24.6	85.9	32.9	36.5	2.0	31.0	35.0	45.6	+9.0
	✓	✓	✓		93.2	56.7	84.1	40.4	37.5	39.5	44.1	35.1	87.1	43.2	80.3	65.8	29.8	87.7	29.6	41.9	0.0	44.4	52.6	52.3	+15.7
	✓	✓		✓	89.0	38.6	80.7	37.1	27.2	42.8	41.5	20.7	85.8	42.4	74.8	64.8	17.8	87.6	30.8	39.4	0.0	41.0	34.6	47.6	+11.0
	✓	✓	✓	✓	91.5	52.4	82.9	42.0	35.7	40.0	44.4	43.3	87.0	43.8	79.5	66.5	31.4	86.7	41.1	52.5	0.0	45.4	53.8	53.7	+17.1
	self distill.	stage 1 init.	sup init.	self-sup init.																				mIoU	gain
stage 2				✓	90.0	57.4	81.8	42.0	40.2	43.8	50.3	50.9	87.6	42.6	80.0	69.2	32.9	87.8	45.5	56.9	0.0	46.0	55.4	55.8	+19.2
	✓	✓			91.4	53.3	83.4	41.3	37.8	43.9	53.0	47.9	88.3	46.1	79.9	70.5	33.2	89.0	48.4	54.6	0.0	50.5	56.7	56.3	+19.7
	✓		✓		91.0	50.2	83.1	40.1	39.8	43.5	51.9	48.1	87.9	45.9	78.5	69.6	34.3	87.9	41.3	56.6	0.0	51.7	57.1	55.7	+19.1
	✓			✓	89.4	56.5	81.3	46.3	42.7	45.1	52.2	51.3	88.5	47.0	82.9	69.3	36.5	87.4	46.0	57.5	0.6	45.9	54.5	56.9	+20.3
stage 3	✓			✓	87.8	56.0	79.7	46.3	44.8	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	59.4	1.0	48.9	56.4	57.5	+20.9

Table 6: Ablation study of each proposed component. The whole training involves three stages, where knowledge distillation can be applied in the last two stages. Here, ST stands for self-training, PD for prototypical denoising, and SL for structure learning.

4. Qualitative comparison

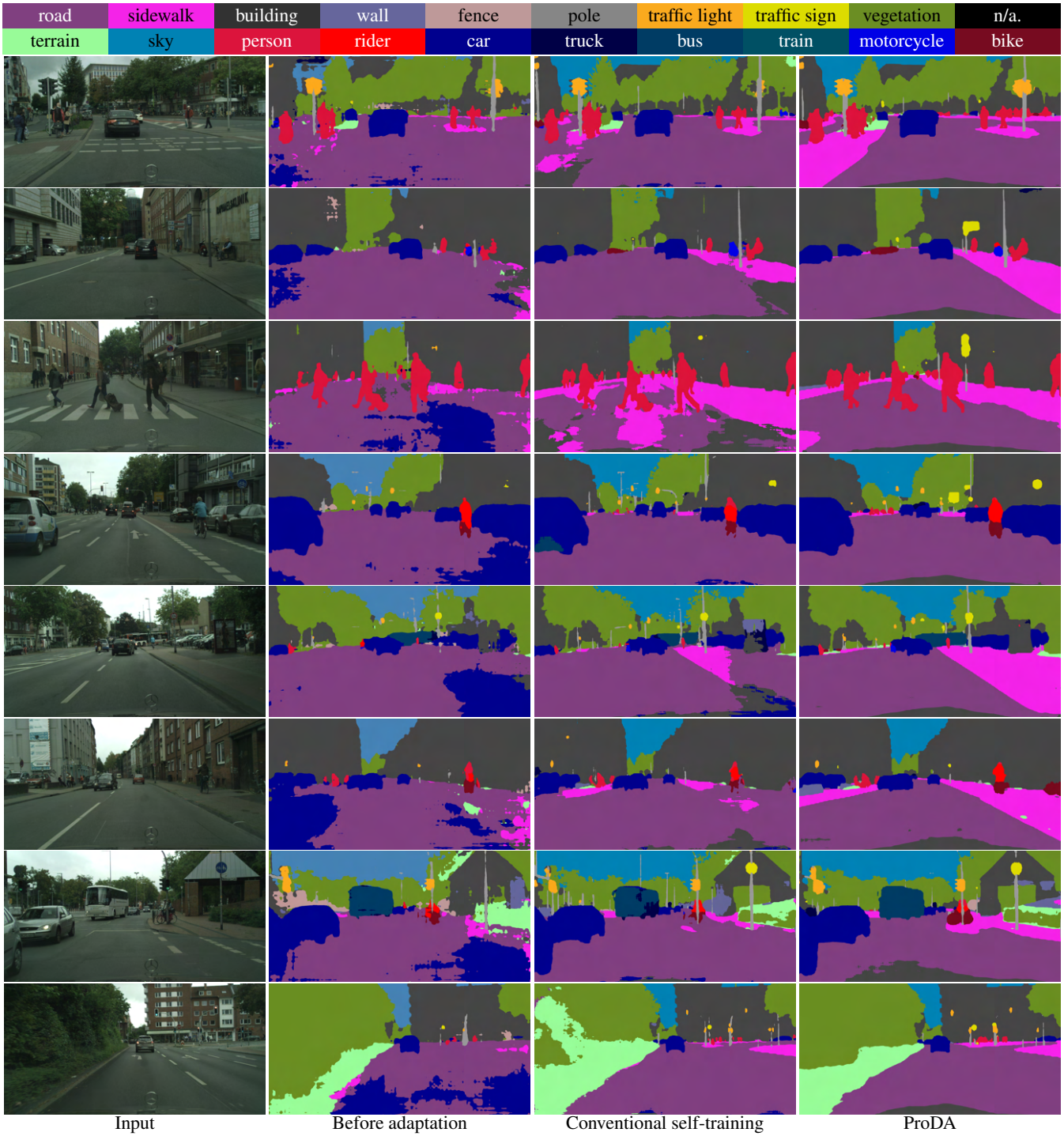

Figure 1: Qualitative results of semantic segmentation on the Cityscapes dataset. From left to right: input, before adaptation, conventional self-training, ProDA.

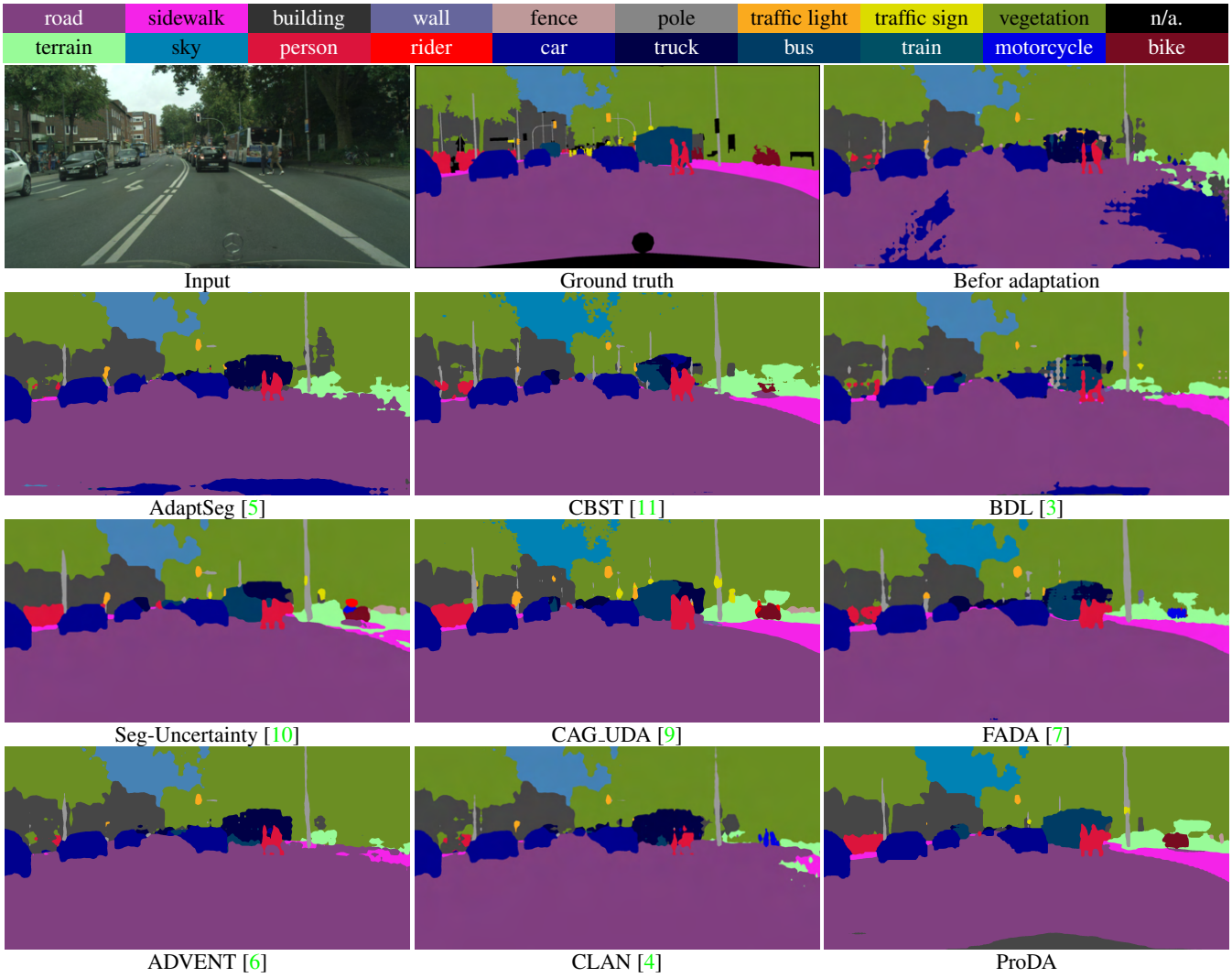

Figure 2: Qualitative comparisons of different methods.

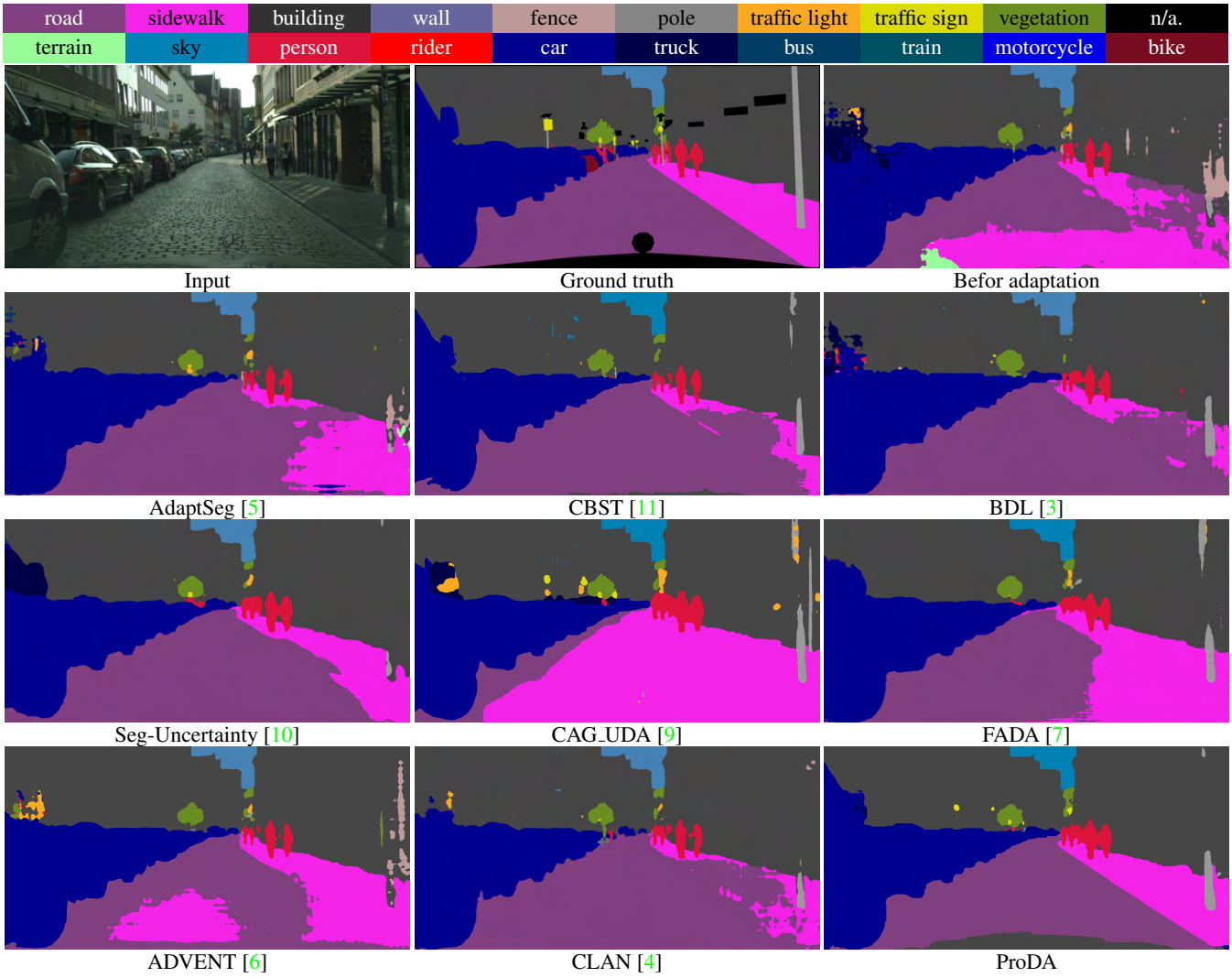

Figure 3: Qualitative comparisons of different methods.

References

[1] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 1

[2] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 1

[3] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6936–6945, 2019. 6, 7

[4] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2507–2516, 2019. 6, 7

[5] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker. Learning to adapt structured output space for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3, 6, 7

[6] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2517–2526, 2019. 6, 7

[7] Haoran Wang, Tong Shen, Wei Zhang, Lingyu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *The European Conference on Computer Vision (ECCV)*, August 2020. 6, 7

[8] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *IEEE International Conference on Computer Vision*, 2019. 1

[9] Qiming Zhang, Jing Zhang, Wei Liu, and Dacheng Tao. Category anchor-guided unsupervised domain adaptation for semantic segmentation. In *Advances in Neural Information Processing Systems*, pages 433–443, 2019. 6, 7

[10] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *arXiv preprint arXiv:2003.03773*, 2020. 6, 7

[11] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 289–305, 2018. 6, 7