Exploiting Domain Properties in Language-Driven Domain Generalization for Semantic Segmentation

Supplementary Material

6. Details of the Baseline Losses

Our baseline [36] exploits two type of losses: $\mathcal{L}_{seg}$ for learning semantic segmentation task learning, $\mathcal{L}_{reg}$ for regularization to maintain visual and textual knowledge of the pretrained model. Specifically, $\mathcal{L}_{seg}$ is formulated as follows:

$$\mathcal{L}_{seg} = \mathcal{L}_{cls} + \lambda_{bce}\mathcal{L}_{bce} + \lambda_{dice}\mathcal{L}_{dice}, \tag{4}$$

where λ_{bce} and λ_{dice} are weight coefficients of their corresponding losses. $\mathcal{L}_{cls}$ is a classification loss for the class predictions $\hat{c}_q$, and both $\mathcal{L}_{bce}$ and $\mathcal{L}_{dice}$ losses are binary cross-entropy loss and dice loss for the mask predictions $\hat{y}^{\text{mask}}$. The outputs of each queries are matched to the ground truth class and mask through the fixed matching.

In addition, $\mathcal{L}_{reg}$ is computed as follows:

$$\mathcal{L}_{reg} = \mathcal{L}^{L}_{reg} + \mathcal{L}^{VL}_{reg} + \mathcal{L}^{V}_{reg}, \tag{5}$$

where $\mathcal{L}^{L}_{reg}$, $\mathcal{L}^{VL}_{reg}$, and $\mathcal{L}^{V}_{reg}$ refers to language regularization, vision-language regularization, and vision regularization, respectively. Each loss is derived as follows:

$$\mathcal{L}^{L}_{reg} = \text{Cross-Entropy}(\text{Softmax}(\hat{t}\hat{T}_0^{\top}, I_K)), \tag{6}$$

$$\mathcal{L}^{VL}_{reg} = \text{Cross-Entropy}(\text{Softmax}(S/\tau), y), \tag{7}$$

$$\mathcal{L}^{V}_{reg} = |v^{\text{CLS}} - v_0^{\text{CLS}}\|_2. \tag{8}$$

Specifically, $\mathcal{L}^{L}_{reg}$ encourages the text feature t to follow a text feature T_0 effective for semantic segmentation task which is obtained with the fixed prompt template 'a clean origami of a $\{\text{class}_k\}$' [31]. The loss matches the cosine-similarity matrix $\hat{t}\hat{T}_0^{\top}$ with the K-dimensional identity matrix I_K via the cross-entropy loss. Secondly, $\mathcal{L}^{VL}_{reg}$ enhances the alignment of the visual feature $v = ENC_I(x)$ and text feature t by matching the score map $S = \hat{v}\hat{t}^{\top}$ with the ground-truth segmentation map y. $\hat{v}$ indicates the normalized visual feature and τ denotes a temperature coefficient. Lastly, $\mathcal{L}^{V}_{reg}$ contributes to preserving visual knowledge of the VLM during training by minimizing the discrepancy between class tokens v^{CLS} and v_0^{CLS} which are obtained from the training backbone and the frozen one, respectively.

Notably, our proposed domain-aware context prompt learning and domain-robust consistency learning are effectively combined with the baseline objectives, significantly improving the overall performance for DGSS task.

7. Hyperparameter Analysis

The quantitative analyses on hyperparameters λ_{contra}, λ_{cons}, λ_{tau} are provided in Tab. 6, which are weighting factors of $\mathcal{L}_{\text{contra}}$, $\mathcal{L}_{\text{cons}}$, and a temperature scaler in $\mathcal{L}_{\text{contra}}$. We note that we empirically set these hyperparameters for the balanced optimization of all training losses. The best performance is obtained when λ_{contra}, λ_{cons}, λ_{tau} are set as 1.0, 5.0, 0.5, respectively. Excessively reducing or increasing the weighting factors resulted in marginal improvements over the baseline. The temperature parameter τ achieved optimal performance at 0.5, adequately reducing the entropy of output distribution in the similarity matrix, thereby facilitating loss convergence.

Models (GTAV)	Parameter	Cityscapes	BDD	Mapillary	Avg.
Baseline	-	57.5	47.66	59.76	54.97
DPMFormer	$\lambda_{\text{contra}} = 0.1$	57.95	49.97	61.03	56.32
	$\lambda_{\text{contra}} = 0.5$	58.22	50.00	61.00	56.41
	$\lambda_{\text{contra}} = 1$	59.00	51.80	63.62	58.14
	$\lambda_{\text{contra}} = 10$	58.21	50.19	61.64	56.68
	$\lambda_{\text{cons}} = 1$	57.54	48.91	60.94	55.80
	$\lambda_{\text{cons}} = 5$	58.10	49.48	61.20	56.26
	$\lambda_{\text{cons}} = 10$	59.00	51.80	63.62	58.14
	$\lambda_{\text{cons}} = 50$	59.08	49.82	62.03	56.98
	$\tau = 0.1$	58.50	50.86	62.53	57.30
	$\tau = 0.5$	59.00	51.80	63.62	58.14
	$\tau = 1$	58.58	50.05	61.94	56.86
	$\tau = 2$	56.81	49.52	60.83	55.72

Table 6. Hyperparameter analysis on synthetic-to-real scenarios with CLIP backbone (ViT-B).

8. Model Performance on Diverse Corruptions

Method	Blur	Noise	Digital	Weather	Elastic Transform	Average
TQDM [36]	39.40	**20.24**	52.56	46.03	**73.50**	42.02
DPMFormer	**40.08**	18.75	**53.57**	**48.85**	73.04	**42.72**

Table 7. Quantitative evaluation on Cityscapes-to-Cityscapes-C with corruption level 5.

We present the model performance on Cityscapes [7]-to-Cityscapes-C [16] with corruption level 5 in Tab. 7 with the CLIP pretrained backbone (ViT-B). We group corruptions into Blur, Noise, Digital, Weather, and Elastic Transform. As described, DPMFormer surpasses another language-driven DGSS method [36] especially against blur, digital, and weather corruptions that induce a large texture changes.

9. Domain Generalization for Image Classification

In Tab. 8, we evaluate DPMFormer on multi-source domain generalization benchmarks [2, 29, 48, 52] with CLIP

Method	PACS	VLCS	Office-Home	Terra
ZS-CLIP [41]	90.7 ± 0.0	80.0 ± 0.0	70.8 ± 0.0	23.8 ± 0.0
CoCoOp [63]	91.9 ± 0.6	81.8 ± 0.3	73.4 ± 0.4	34.1 ± 3.0
DPL [60]	91.8 ± 0.7	80.8 ± 0.8	73.6 ± 0.4	34.4 ± 1.0
SPG [1]	**92.8 ± 0.2**	**84.0 ± 1.1**	73.8 ± 0.5	**37.5 ± 1.8**
DPMFormer	91.5 ± 0.3	81.5 ± 1.0	**73.9 ± 0.4**	35.0 ± 2.1

Table 8. Comparisons on image classification DG methods.

ResNet50 backbone. Following conventions, the evaluation is conducted in the leave-one-domain-out manner and we report the average domain accuracy of the model selected using the training-domain validation set method. In summary, DPMFormer achieves performance comparable to previous prompt learning methods for image classification [1, 60]. In particular, we achieve the best performance on Office-Home [52], and surpass CoCoOp [63] and DPL [1] on Terra-Incognita [2].

We note that existing prompt learning studies for image classification [1, 60] are not suitable for the single-source setting of DGSS, as they either require multiple source datasets [60] or depend on multi-stage training and adversarial learning [1] to obtain prompts. In contrast, our domain-aware prompt generation requires image transformations and a contrastive objective, making it more effective for semantic segmentation tasks.

10. Computational Overhead

With DPMFormer, each training iteration takes 1.712 seconds, slightly more than the baseline's 1.501 seconds. Meanwhile, its inference time of 1.376 seconds per batch remains comparable to 1.004 seconds of the baseline. The batch size is reduced by half due to texture perturbation, but still maintains better performance under the same training setting.

11. Class-wise Quantitative Comparison

Through Tab. 9 to Tab. 13, we compare class-wise IoU of DPMFormer with TQDM [36] with the CLIP initialized models. Noticeably, DPMFormer demonstrates higher performance in most classes in various scenarios and shows comparable score even in other cases. In summary, the average IoU consistently outperforms the competitor, verifying the superiority of DPMFormer in DGSS.

12. Precision-Recall Curve Comparison

In Fig. 6 and 7, we depict Precision-Recall curves of each class with Averaged Precision (AP) in synthetic-to-real scenario (GTA [42]-to-BDD [58]) with CLIP (ViT-B) and EVA02-CLIP backbones, respectively. Compared to TQDM [36], DPMFormer shows better performance in most classes, validating the effectiveness of domain-aware context prompt learning as well as consistency learning.

13. Limitation and Future Work

Domain-aware context prompt learning utilizes the global representation of the frozen backbone to obtain domain-specific properties of the image. However, some local textures may differ from the global textures in complex scenes. For example, in night driving scene, the roads are brightened due to car headlights, whereas the sky and surroundings are darkened. Hence, exploiting local texture patterns for more detailed prompt generation can be a good initial motivation for future language-driven DGSS. In addition, the design of the domain-aware prompt generator h_θ and textural perturbations can be further advanced to accomplish better performance. Meanwhile, the unshared label space between the source and the target domain hinders the model from correctly interpreting the image context. From this perspective, we believe addressing DGSS through open-set domain adaptation and the integration of VLM should be a promising direction for future research.

14. Additional Qualitative Results

Through Fig. 8 to Fig. 12, we provide additional qualitative results of DPMFormer in various scenarios. DPMFormer consistently yields more accurate segmentation results than TQDM [36] in scenes under diverse environments and from various locales.

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic Light	Traffic Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Avg
TQDM	**90.97**	**50.91**	**88.24**	36.52	38.54	**47.76**	54.82	45.78	89.10	**40.78**	**89.67**	74.33	40.46	85.73	39.41	**60.69**	**46.71**	29.85	47.84	57.79
Ours	87.92	46.60	88.19	**38.83**	**39.37**	47.27	**54.90**	**49.24**	**89.11**	40.49	89.52	**74.90**	**43.08**	**88.11**	**54.42**	55.95	35.72	**44.16**	**53.15**	**59.00**

Table 9. Class-wise quantitative comparison (IoU) in synthetic-to-real (GTA [42]-to-Cityscapes [7]) scenario with the CLIP-pretrained ViT-B backbone.

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic Light	Traffic Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Avg
TQDM	88.76	48.75	79.85	22.46	30.48	**41.94**	45.72	39.35	75.04	40.69	88.11	58.43	26.54	80.00	32.39	43.10	0.00	44.99	33.26	48.41
Ours	**90.79**	**49.72**	**81.38**	**29.56**	**34.65**	41.68	**47.03**	**42.66**	**75.82**	**42.24**	**88.30**	**59.72**	**32.23**	**84.04**	**37.49**	**61.20**	0.00	**50.12**	**35.61**	**51.80**

Table 10. Class-wise quantitative comparison (IoU) in synthetic-to-real (GTA [42]-to-BDD [58]) scenario with the CLIP-pretrained ViT-B backbone.

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic Light	Traffic Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Avg
TQDM	89.93	54.28	85.24	41.74	43.44	**51.80**	**56.93**	67.24	79.36	50.58	94.27	**75.94**	**56.34**	86.62	51.80	54.65	19.76	57.79	41.03	60.99
Ours	**90.20**	**58.33**	**85.42**	**43.19**	**45.21**	51.57	56.36	**68.49**	**79.91**	**51.64**	**94.39**	75.28	56.10	**88.70**	**59.99**	**61.80**	**32.77**	**62.89**	**46.65**	**63.35**

Table 11. Class-wise quantitative comparison (IoU) in synthetic-to-real (GTA [42]-to-Mapillary [34]) scenario with the CLIP-pretrained ViT-B backbone.

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic Light	Traffic Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Avg
TQDM	92.55	56.65	83.66	24.37	33.93	42.44	46.87	48.22	**84.02**	**45.68**	**93.80**	58.51	28.42	86.90	38.87	38.8	0.27	37.23	26.90	50.95
Ours	**92.76**	**57.68**	**83.83**	**28.73**	**39.57**	**45.04**	**49.91**	**51.20**	83.96	44.53	93.78	**62.57**	**40.21**	**87.37**	**40.15**	**47.95**	**0.29**	**53.07**	**38.78**	**54.81**

Table 12. Class-wise quantitative comparison (IoU) in real-to-real (Cityscapes [7]-to-BDD [58]) scenario with the CLIP-pretrained ViT-B backbone.

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic Light	Traffic Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Avg
TQDM	**90.66**	**53.19**	**87.34**	48.39	54.05	51.64	58.57	73.51	83.93	53.16	94.64	74.39	61.32	89.62	58.58	56.80	21.92	58.84	56.87	64.60
Ours	90.65	52.88	86.99	**48.84**	**57.20**	**54.03**	**61.57**	**76.29**	**88.21**	**53.30**	**97.10**	**77.04**	**65.38**	**90.45**	**62.03**	**66.30**	**24.87**	**68.04**	**65.51**	**67.72**

Table 13. Class-wise quantitative comparison (IoU) in real-to-real (Cityscapes [7]-to-Mapillary [34]) scenario with the CLIP-pretrained ViT-B backbone.

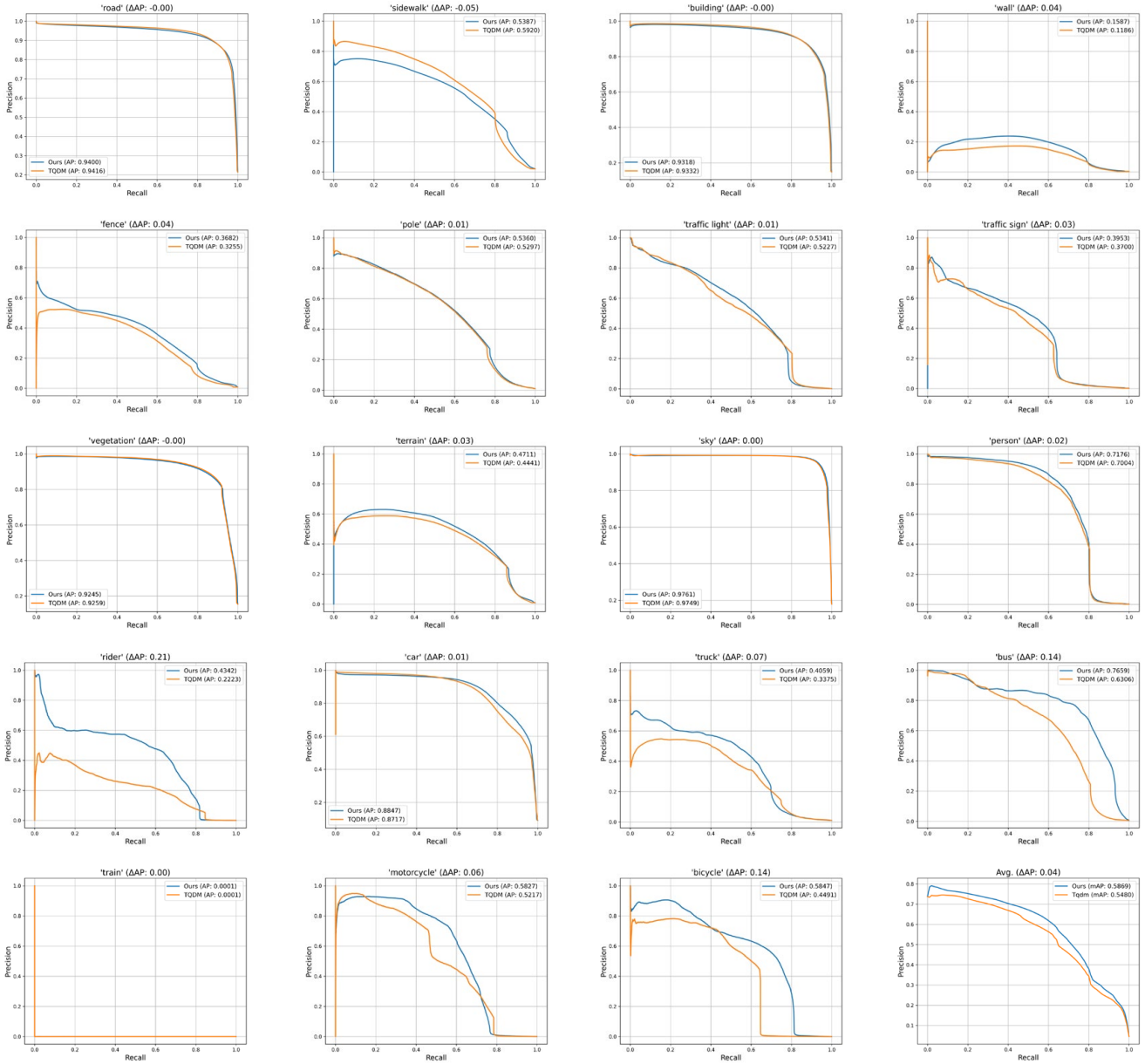

Figure 6. Precision-Recall curve and Average Precision (AP) on synthetic-to-real scenario (GTA [42]-to-BDD [58]) with the CLIP-pretrained backbone (ViT-B).

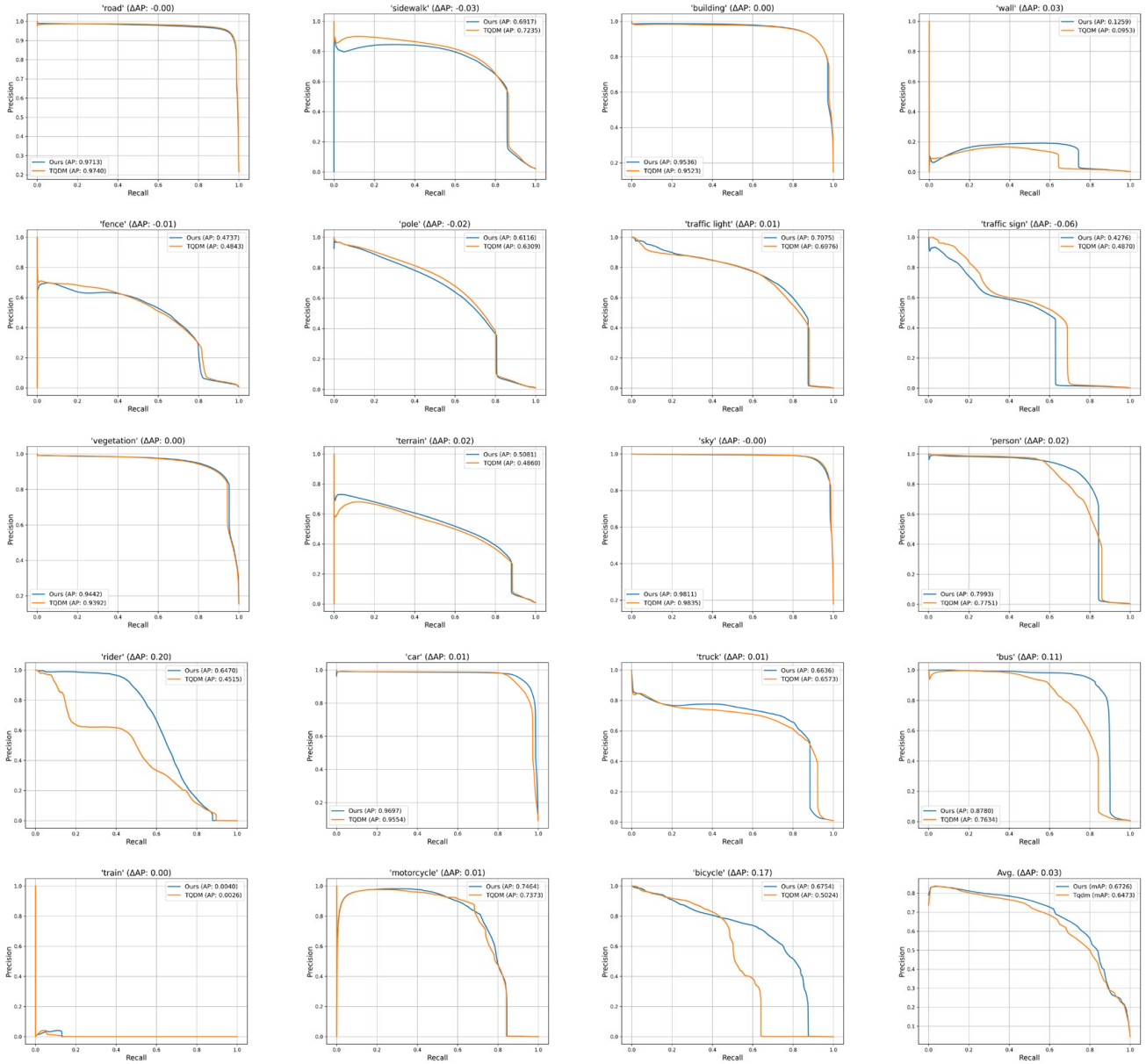

Figure 7. Precision-Recall curve and Average Precision (AP) on synthetic-to-real scenario (GTA [42]-to-BDD [58]) with the EVA02-CLIP [45] pretrained backbone.

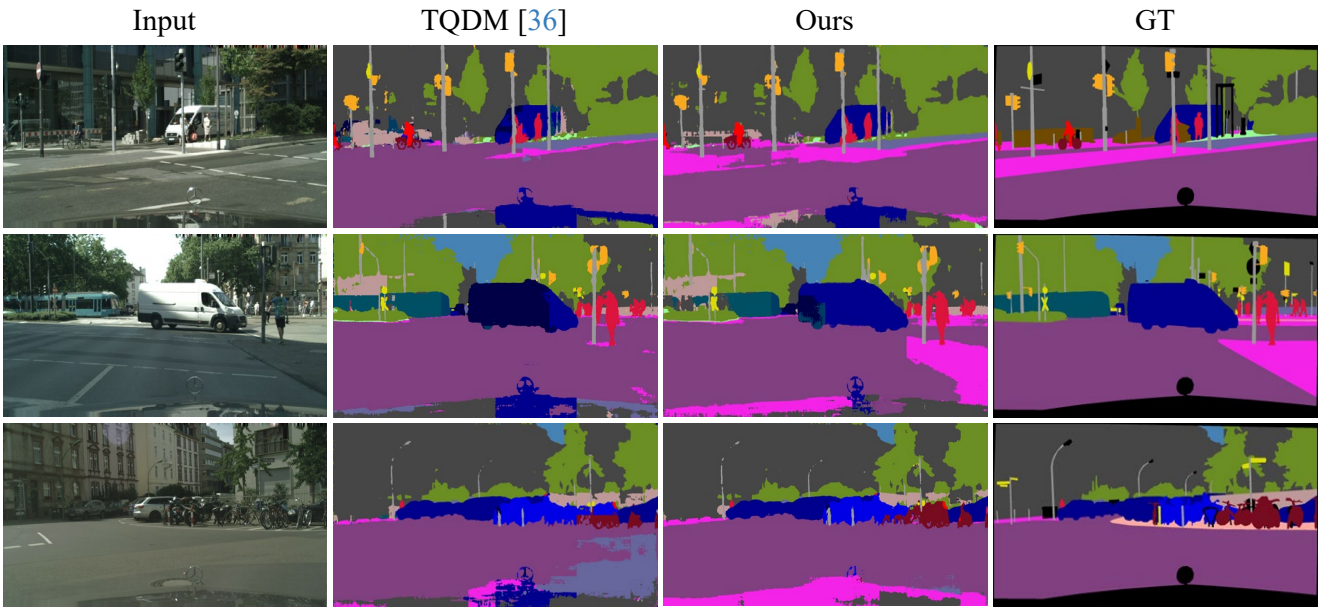

Figure 8. Qualitative comparison on synthetic-to-real scenario (GTA [42]-to-Cityscapes [7]) with the CLIP-pretrained backbone (ViT-B). With the first image, TQDM [36] mispredicts sidewalk as roads and shows confusion on the region next to the bicycle rider. TQDM also confuses the car with the truck (second row) and misclassify bicycles as motorcycles (third row). On the other hand, DPMFormer produces more reliable and accurate segmentation results in these scenes.

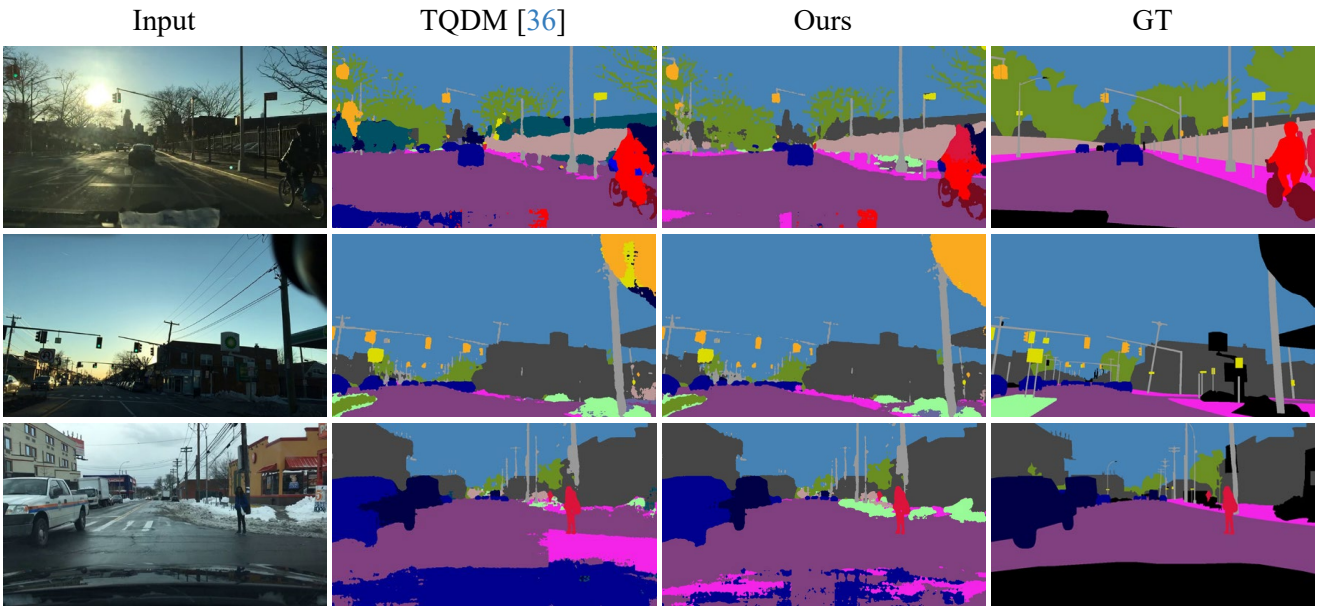

Figure 9. Qualitative comparison on synthetic-to-real scenario (GTA [42]-to-BDD [58]) with the CLIP-pretrained backbone (ViT-B). Due to the large illumination contrast caused from the intense sunlight (first row), TQDM [36] wrongly mark the buliding as a train. In addition, TQDM perplexes the road as 'car' and 'sidewalk' due to their textural similarity. Contrarily, DPMFormer shows consistent performance under various environments, almost reaching ground-truth segmentation maps.

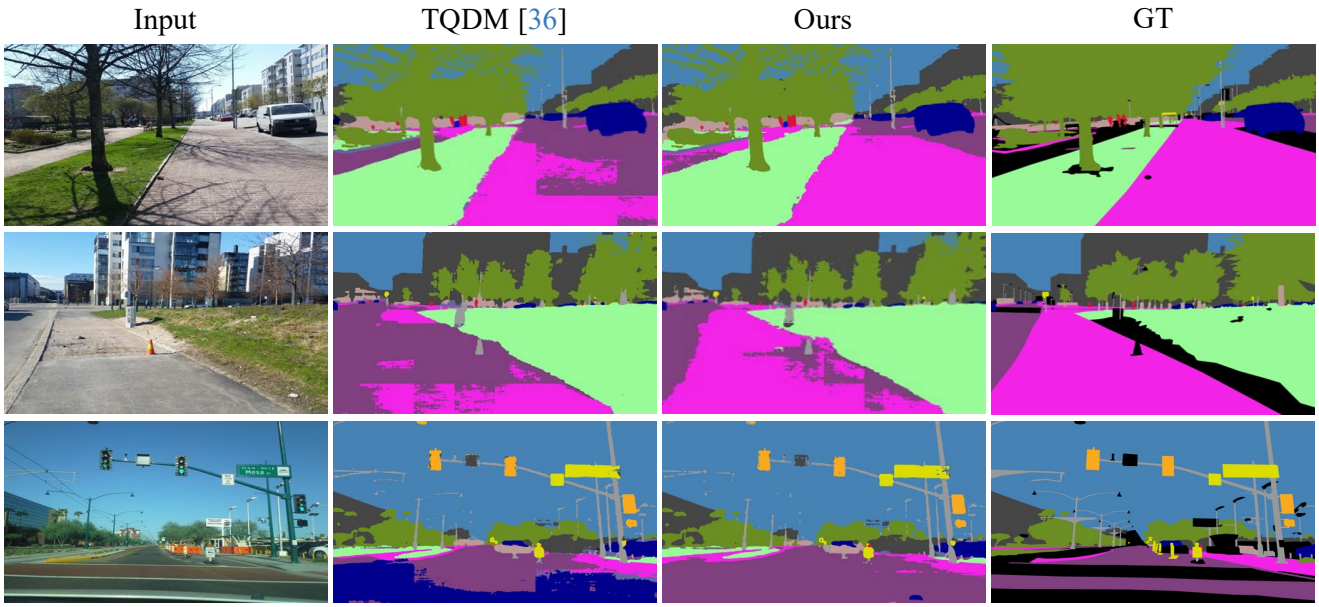

Figure 10. Qualitative comparison on synthetic-to-real scenario (GTA [42]-to-Mapillary [34]) with the CLIP-pretrained backbone (ViT-B). TQDM [36] confounds road as sidewalk or car because of the textual changes gap from the synthetic texture. Conversely, DPMFormer predicts accurately by utilizing domain-aware context prompt and the domain-robust cues learned from consistency losses.

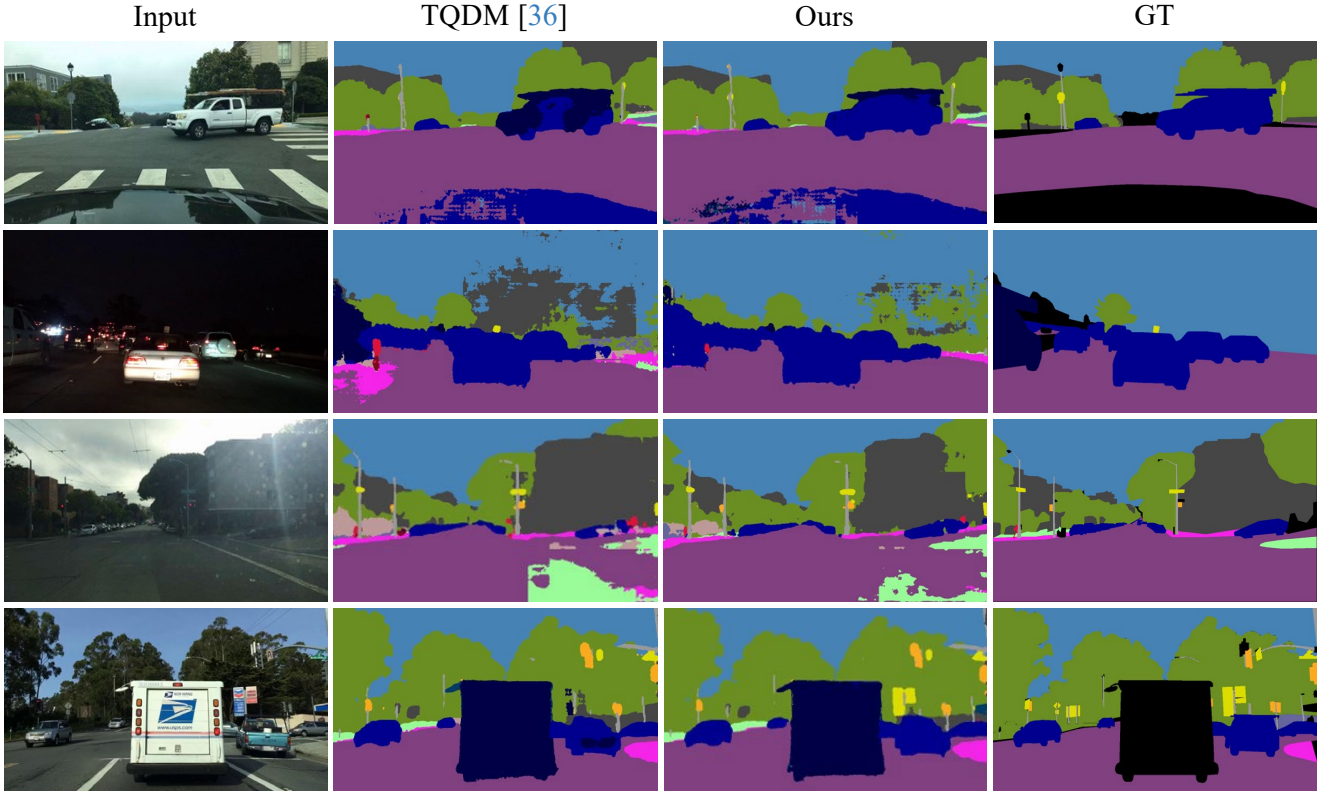

Figure 11. Qualitative comparison on real-to-real scenario (Cityscapes [7]-to-BDD [58]) with the CLIP-pretrained backbone (ViT-B). In the first image, TQDM mispredicts the car and the bus due to the occlusion. In case of the nighttime (second row) and the daytime (third row) scenes, predictions gets noisy due to the textural ambiguity. As shown in the last row, TQDM fails to catch traffic signs which have different design from the Cityscapes dataset. DPMFormer demonstrates its efficacy by producing more precise segmentation results compared to TQDM.

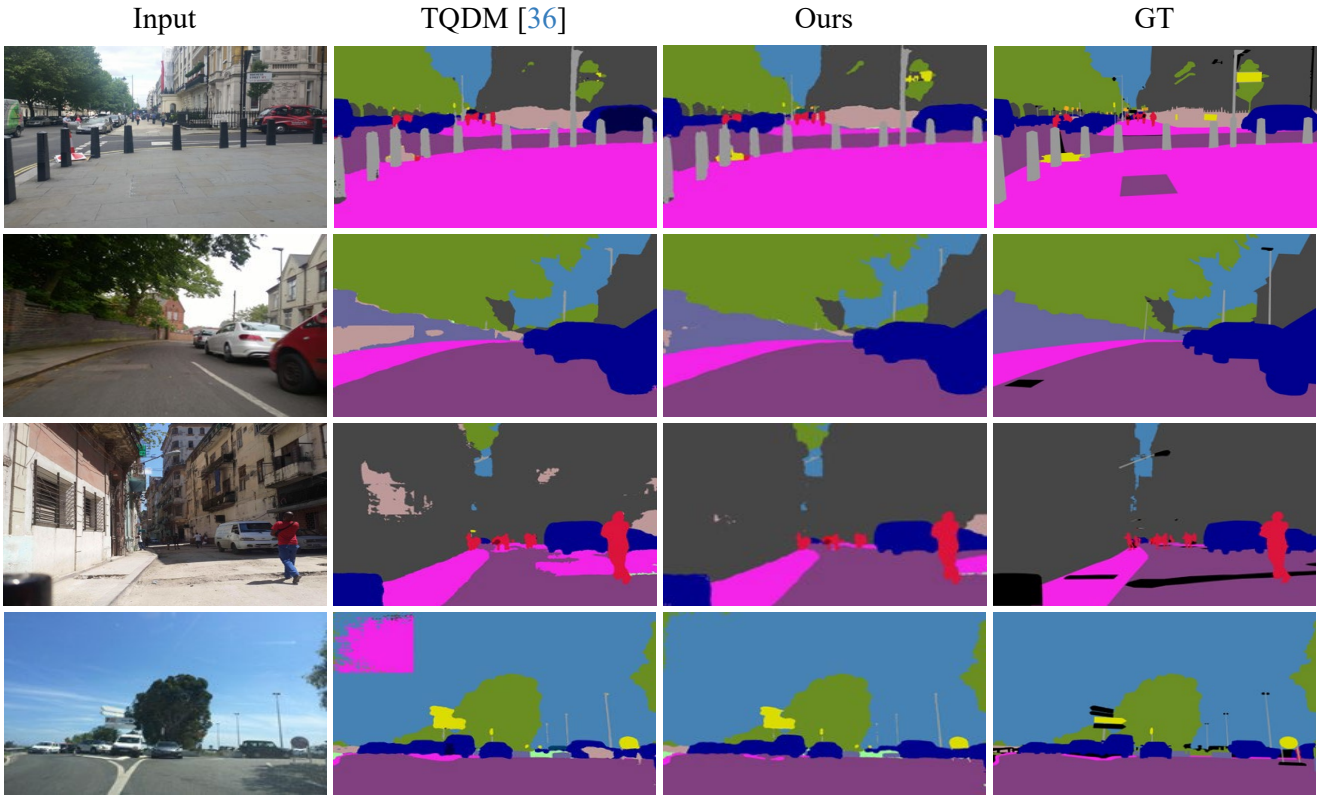

Figure 12. Qualitative comparison on real-to-real scenario (Cityscapes [7]-to-Mapillary [34]) with the CLIP-pretrained backbone (ViT-B). Due to the location difference between the datasets, TQDM miss traffic signs (first row) and misclassify the walls (second row) and the road (third row). Also in the clean daytime image (fourth row), fallacious predictions are observed in the sky and infront of the car on the right side. On the other hand, DPMFormer generates clean and reliable predictions among these images.