Dynamically Instance-Guided Adaptation: A Backward-free Approach for Test-Time Domain Adaptive Semantic Segmentation (Supplementary Material)

Wei Wang[1] , Zhun Zhong[2], Weijie Wang[2], Xi Chen[3] , Charles Ling[1], Boyu Wang[1*], Nicu Sebe[2]
[1]Western University [2]University of Trento [3]Huawei Noah's Ark Lab

In this supplementary material, we present a comprehensive analysis of our proposed method along with additional experimental results and discussion.

Firstly, in Sec. A, we provide a detailed ablation study of the proposed DAM and SAM modules to demonstrate their effectiveness in improving the segmentation performance.

Secondly, we investigate the effectiveness of applying the DIGA method on state-of-the-art models in Sec. B. The results show that DIGA can further improve the segmentation performance of these models.

Furthermore, we conduct a sensitivity analysis of the hyperparameters in Sec. C to provide insights into the impact of these parameters on the segmentation performance.

Lastly, we discuss the potential of incorporating transformers into the proposed method and present an additional baseline for comparison in Sec. D. This section also highlights potential future directions for research in this field.

A. Further Ablation Studies of DAM and SAM Module

In this section, we present an in-depth ablation study of the DAM and SAM modules. We begin by examining the necessity of the DAM module for the SAM module. Specifically, we compare the performance of DAM+SAM with that of SAM only. The results, as presented in Tab. 4, demonstrate that DAM+SAM consistently outperforms SAM only on all target domains, with an average improvement of 2.47%. These findings suggest that the DAM module can consistently enhance the performance of the SAM module. Next, we investigate whether SAM contributes to the performance of DAM. While the quantitative analysis is reported in the experiment section of the main paper, we provide a qualitative illustration in Fig. 6. This figure depicts two people riding on the same bicycle, which is a common occurrence in the real world, but not present in the synthetic source GTA5. From the figure, it is apparent that DAM alone struggles to distinguish between the people and the bicycles. However, with the incorporation of SAM, the results are significantly improved, providing evidence for the effectiveness of the SAM module.

*Corresponding author

Table 4. SAM performance with or without DAM. Source domain: GTA5. ↑: with improvement.

Method	CS	BDD	MA	IDD	CC	Avg.
SAM w/o DAM	42.79	33.40	40.80	42.28	30.70	37.99
SAM w/ DAM	**45.81**	**35.78**	**44.25**	**42.73**	**33.72**	**40.46**↑

B. Applying SAM on State-of-the-art Methods

In this section, we explore the potential of integrating the SAM module with other state-of-the-art TTDA methods. The SAM module's plug-in design makes it a seamless addition to existing methods, as it can use the model's feature to predict the confidence of its output and enhance the prediction accordingly. By combining the predictions of the model and the SAM module, the final prediction can be obtained.

To assess the compatibility of DIGA with other methods, we present the adaptation results of IN, DUA, and SITA with and without SAM modules from GTA5 to five target domains in Table 7. Our results demonstrate that the use of SAM modules can significantly improve the performance of IN, DUA, and SITA, with an average improvement of 4.61%, 2.17%, and 3.31%, respectively. Although there are some exceptions, such as the BDD domain, where applying the SAM module does not lead to a performance improvement. We speculate that this could be due to less significant semantic shifts between GTA and BDD compared to other domains.

C. Sensitivity Analysis of Hyperparameters

The sensitivity of hyperparameters is a critical issue in the training of deep neural networks, where small variations in hyperparameters can significantly impact the model's performance. This issue is exacerbated in TTDA since hyperparameter tuning is not possible during the adaptation

Figure 6. An illustration example of the effectiveness of SAM. GTA5→IDD.

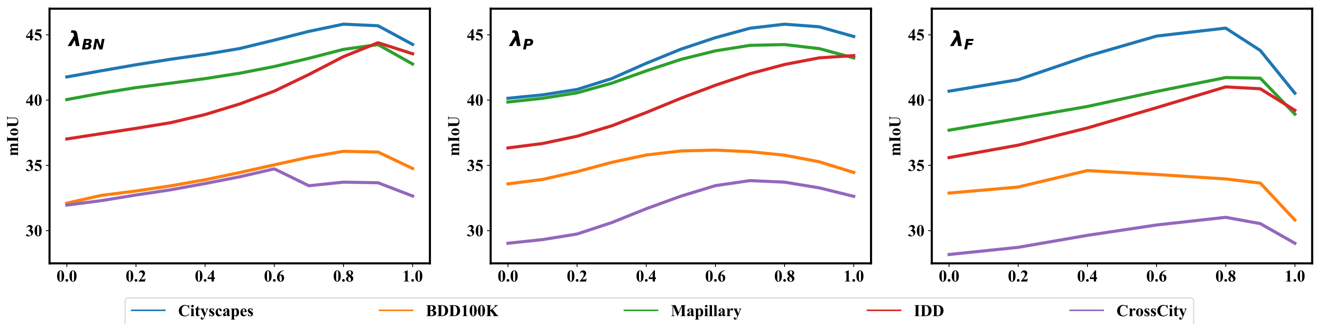

Figure 7. Parameter sensitivity analysis. From left to right, λ_{BN}, λ_P, λ_F. Source: GTA5.

Table 5. Applying SAM on state-of-the-art methods. Source domain: GTA5. ↑: with improvement.

Method	CS	BDD	MA	IDD	CC	Avg.
IN	34.25	29.64	35.01	29.80	23.87	30.51
IN+SAM	40.08	28.66	37.66	38.04	31.14	35.12↑
DUA	37.79	31.76	40.26	34.75	26.32	34.18
DUA+SAM	42.25	28.66	39.17	41.10	30.55	36.35↑
SITA	40.64	32.94	37.80	35.66	28.19	35.26
SITA+SAM	42.99	32.91	41.02	42.43	33.49	38.57↑

process. As a result, unstable hyperparameters can lead to unpredictable and undesirable adaptation performance, which is not desirable in practical applications.

To address this issue, we conduct a series of experiments to analyze the sensitivity of TTDA to hyperparameters in the context of segmentation. We focus on the most significant and distinctive hyperparameters of our proposed Domain-Invariant Graph-based Adaptation (DIGA) model: λ_{BN} in the domain adaptation module (DAM), λ_P in the semantic adaptation module (SAM), and λ_F in the final merging process.

The λ_{BN} hyperparameter balances the trade-off between instance-level and historical statistics in the batch normalization layer, while the λ_P hyperparameter balances the trade-off between instance-level and historical statistics in the prediction layer. The λ_F hyperparameter balances the trade-off between direct prediction and prototype-based prediction in the final merging process.

To evaluate the impact of these hyperparameters on TTDA performance, we conduct experiments on five different target domains, and we vary the values of each hyperparameter between 0 and 1 with a step size of 0.1. Figure 7 shows the performance of TTDA with different hyperparameters. We observe that the optimal values of λ_{BN}, λ_P, and λ_F consistently cluster around 0.8 for all target domains, indicating that these hyperparameters are not highly sensitive to the target domain.

Moreover, we find that setting the hyperparameters to 0.8 consistently achieves better performance than setting them to 0 or 1 across almost all datasets. This result suggests that incorporating both instance-level and historical statistics in the DAM and SAM modules can improve performance and that direct prediction and prototype-based prediction can complement each other in the final merging process. Importantly, we use the same hyperparameters for all source domains, rather than selecting domain-specific hyperparameters, which demonstrates the robustness of our method.

D. Discussions

Transformers. Transformers have become increasingly popular in computer vision tasks, such as image classification [1] and object detection [5]. Recently, studies have shown that transformers can achieve state-of-the-art performance on semantic segmentation tasks [9]. To facilitate future comparisons, we provide the basic results of the Segformer model [9] in Table 6. The results show that Segformer achieves an average performance of 35.90%, and applying a spatial attention module (SAM) can further improve its performance to 41.62%.

Unfortunately, the design of the DAM module in DIGA cannot be directly applied to transformer-based models due to the absence of batch normalization (BN) layers. Consequently, SAM can only work on the shifted features after being processed by the networks. This limitation also applies to many other TTDA methods that rely on BN layers [3, 4]. Therefore, it is reasonable to speculate that designing a transformer-appliable DAM could further improve the performance of the model, which may be a topic for future work.

Table 6. The average performance of source model trained with GTA5 on five target domains. ↑: with improvement.

Method	Average mIoU.
Segformer	35.90
Segformer+DIGA	41.62↑

Additional baselines. We present additional baselines in Table 7 for comparison purposes. It is worth noting that this comparison is not entirely fair, as DA methods have access to the source domain during the training process, while the test is conducted after adaptation on the validation set. Nevertheless, our proposed method achieves superior results when compared to AdaptSeg and AdvEnt by 3.4% and 0.3%, respectively, even without accessing the source domain. On the other hand, more recent DA works such as ProDA and DACS demonstrate better performance than TTDA methods, with top-2 performances of 57.5% (ProDA) and 52.1% (DACS), respectively. This is expected since the test-time adaptation is conducted under restricted conditions. The large performance gap suggests that there is still room for further exploration in the adaptation space, leaving an opportunity for future work.

Table 7. Additional baselines for future studies. Source domain: GTA5. Target domain: Cityscapes.

Method	Setting	Published	mIoU.
DACS [6]	DA	WACV'2021	52.1
ProDA [2]	DA	CVPR'2021	57.5
AdaptSeg [7]	DA	CVPR'2018	42.4
AdvEnt [8]	DA	CVPR'2019	45.5
Ours	TTDA	-	45.8

References

[1] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *NIPS*, 2021. 2

[2] Xinyue Huo, Lingxi Xie, Hengtong Hu, Wengang Zhou, Houqiang Li, and Qi Tian. Domain-agnostic prior for transfer semantic segmentation. In *CVPR*, 2022. 3

[3] Ansh Khurana, Sujoy Paul, Piyush Rai, Soma Biswas, and Gaurav Aggarwal. Sita: Single image test-time adaptation. *arXiv preprint arXiv:2112.02355*, 2021. 3

[4] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018. 3

[5] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *CVPR*, pages 3611–3620, 2021. 2

[6] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *WACV*, 2021. 3

[7] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *CVPR*, 2018. 3

[8] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019. 3

[9] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *NIPS*, 2021. 2