Learning Cross-View Object Correspondence via Cycle-Consistent Mask Prediction

Supplementary Material

A. Organization

This document contains the following sections:

- More training details are provided in Section B.
- Test-time training on HANDAL-X is provided in Section C.
- More ablation study is provided in Section D.
- Efficiency analysis is provided in Section E
- More qualitative results are provided in Section F.
- Limitation and future work are provided in Section G.

All blue-highlighted rows in the tables denote the default configurations of our method. We refer to test-time training as **TTT** throughout the paper.

B. More Training Details

We train our model using the AdamW optimizer [3] with a cosine learning rate schedule and linear warm-up. We use a batch size of 16. The image size is 512×512. The training process on Ego-Exo4D [2] consists of two stages. In the first stage (*linear probing*), we freeze the two DINOv3 [4] backbones and train the remaining modules for 64K iterations. The learning rate decays from the maximum value of 1×10^{-3} to a minimum of 1×10^{-4}. In the second stage, all parameters are unfrozen and optimized for 640K iterations. The learning rate decays from the maximum value of 1×10^{-5} to a minimum of 1×10^{-6}. To address GPU memory limitations (40GB), we adopt gradient accumulation with a step size of 16, resulting in an effective number of parameter updates of 704K / 16 = 44K. The training process takes approximately 72 hours on 8 NVIDIA RTX A800 GPUs. We maintain an exponential moving average (EMA) of the model parameters throughout training, and use the EMA model as the final model for evaluation. For visibility prediction, we fine-tune only the CLS Head for 96K iterations, which takes approximately 1 hour on the same hardware, using the same training setup as the main binary segmentation task. In the TTT stage, the adaptation takes approximately 3 hours for Ego2Exo and 12 hours for Exo2Ego.

On HANDAL-X, we train for 10 epochs with the learning rate decaying from the maximum value of 2×10^{-4} to a minimum of 2×10^{-6}. The offline training stage requires approximately 2 hours, and the TTT stage requires an additional 1 hour.

Table 1. **Evaluation results on the HANDAL-X benchmark.**

Method	Fine-tuning Datasets	IoU↑
XSegTx [2]	∅	1.5
SEEM [7]	∅	2.5
PSALM [6]	∅	14.2
PSALM [6]	Ego-Exo4D	39.9
ObjectRelator [1]	Ego-Exo4D	42.8
Ours (w/o TTT)	Ego-Exo4D	78.8
Ours	Ego-Exo4D	**80.6**
PSALM [6]	Ego-Exo4D, HANDAL-X	83.4
ObjectRelator [1]	Ego-Exo4D, HANDAL-X	84.7
Ours (w/o TTT)	Ego-Exo4D, HANDAL-X	85.0
Ours	Ego-Exo4D, HANDAL-X	**85.3**

C. Test-time Training on HANDAL-X

Since our results on the HANDAL-X benchmark [1] without TTT already surpass all baselines, we omit the TTT results from the main text. Table 1 presents the quantitative performance with TTT on HANDAL-X, further demonstrating its effectiveness and generalization across benchmarks. We observe that when the baseline IoU is already very high, TTT yields only marginal improvements. The corresponding qualitative results are provided in Figure 4.

D. More Ablation Study

Mask Prediction Method. To enable the model to adaptively predict segmentation masks conditioned on given object features, we further explore an alternative implementation named *Cosine Prediction* for mask generation. The final segmentation mask $\hat{M}_t$ is predicted using both visual tokens and the updated condition token y_{cdt}. Specifically, for the i-th visual token y_i, the prediction is computed as:

$$\hat{M}_t^i = \texttt{Sigmoid}\left(\tau \cdot \texttt{Cos}(y_{\text{cdt}}, y_i) - \beta\right), \tag{1}$$

where $\texttt{Cos}(\cdot,\cdot)$ denotes cosine similarity, and τ and β are learnable temperature and bias parameters as in [5]. They are initialized to 10 and 5, respectively.

Table 2 presents an ablation study comparing the proposed variant with our original method. The results demonstrate that direct mask prediction yields better performance than predicting masks conditioned on object features.

Table 2. **Ablation on the mask prediction method.**

Method	Ego-IoU↑	Exo-IoU↑	mIoU↑
Ours	**41.95**	**47.18**	**44.57**
Cosine Prediction	40.29	46.75	43.52

Dice weight. We investigate the influence of the Dice Loss weight λ_{dice} in our mask supervision objective $\mathcal{L}_{\text{mask}}$. The Dice Loss $\mathcal{L}_{\text{dice}}$ plays an essential role, particularly in scenarios where the target occupies only a small region of the spatial mask, as it effectively addresses class imbalance and encourages better alignment of predicted and ground-truth masks. Table 3 presents a detailed ablation of performance across different values of λ_{dice}. We find that setting $\lambda_{\text{dice}} = 5$ yields the best overall performance, outperforming all other configurations. Notably, when $\lambda_{\text{dice}} = 0$, which effectively removes the Dice Loss, the performance drops significantly across all metrics, confirming the importance of including $\mathcal{L}_{\text{dice}}$. These results highlight the importance of balancing the Dice component within the mask loss.

Table 3. **Ablation of dice weight.**

λ_{dice}	Ego-IoU↑	Exo-IoU↑	mIoU↑
0	28.22	32.76	30.49
0.5	41.11	46.24	43.68
1	41.84	47.17	44.51
2	41.33	46.70	44.02
5	**41.95**	**47.18**	**44.57**
10	41.54	46.72	44.13

Gradian Update Steps of TTT. It is notable that for TTT on Ego-Exo4D, we update for T=2 steps in Ego2Exo but T=6 steps in Exo2Ego. To justify this choice, we conduct an ablation study, and the results are presented in Table 4 and Table 5. We observe that Ego2Exo achieves its best performance with only 2 update steps, after which further updates cause slight degradation. In contrast, Exo2Ego continues to benefit from additional updates and reaches its peak performance at 7 or more steps. Considering the tradeoff between efficiency and performance, we adopt T=6 steps as our default setting. These findings highlight the importance of tuning the number of update steps for each direction individually, as the two tasks differ in their underlying object-size distributions and consequently in their adaptation behavior.

Table 4. **Ablation of TTT steps (Ego2Exo).**

Steps	Ego-IoU↑
1	41.91
2	**41.95**
3	41.90
4	41.88
5	41.84

Table 5. **Ablation of TTT steps (Exo2Ego).**

Steps	Exo-IoU↑
3	46.81
4	46.98
5	47.09
6	47.18
7	**47.23**

Table 6. **Ablation of fine-tuning layers (Ego2Exo).**

Layers	Ego-IoU↑
3	41.90
4	**41.95**
5	41.94
6	41.93
7	41.87

Table 7. **Ablation of fine-tuning layers (Exo2Ego).**

Layers	Exo-IoU↑
8	47.06
9	47.14
10	47.17
11	**47.18**
12	47.14

Fine-tuning Layes of TTT. It is notable that for TTT on Ego-Exo4D, we update the last K=4 layers in Ego2Exo but K=11 layers in Exo2Ego. To justify this choice, we conduct an ablation study, and the results are reported in Table 6 and Table 7. We observe that Ego2Exo achieves its best performance when only a small number of layers are adapted, while deeper adaptation yields diminishing returns or slight degradation. In contrast, Exo2Ego benefits from updating a substantially larger portion of the network, with performance peaking at 11 layers. These findings suggest that Ego2Exo requires only lightweight adjustments for effective adaptation, whereas Exo2Ego demands broader model capacity to accommodate the larger cross-view domain gap.

E. Effeciency Analysis

We agree that performance–latency trade-offs better reflect practical deployment than a single inference-time number. Accordingly, we provide an efficiency analysis in Figure 1, reporting mIoU as a function of inference time by varying the number of test-time optimization steps (from 0 to 1, 2, and beyond). As shown, most of the performance gain is achieved with only 2 gradient updates, while further updates bring diminishing returns. This indicates that our method can achieve improvement with limited additional latency.

F. More Qualitative Results

We provide additional qualitative results on the Ego-Exo4D correspondence benchmark in Figure 2 and Figure 3. We present diverse examples that cover all six scenarios: cooking, health, bike repair, music, basketball, and soccer. The

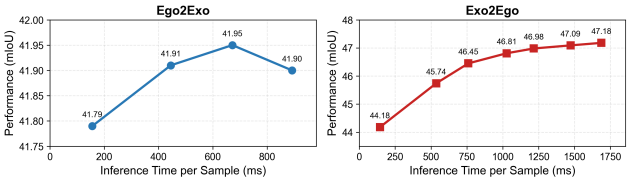

Figure 1. **Performance–latency trade-off under test-time training.**

cooking scenario occupies three rows in each figure because it constitutes the largest portion of the benchmark. Across all scenarios, our method consistently produces masks that closely match the ground truth annotations, demonstrating strong robustness to variations in scene context, object category, occlusion, and viewpoint. These results also illustrate the effectiveness of TTT, which enables the model to focus more accurately on the target object while suppressing visually similar distractors and to generate masks that more completely cover the ground truth regions.

We further present qualitative results on HANDAL-X in Figure 4, illustrating six examples spanning diverse hand–object interaction categories and highlighting the effectiveness of TTT. Our method successfully recovers the target masks in most cases.

G. Limitation and Future Work

All qualitative results are presented without excluding failure cases, providing a comprehensive view of the model's potential errors. We summarize the common failure patterns, from frequent to rare:

- Incomplete coverage of the ground-truth regions.
- Attraction to objects visually similar to the target object in the scene.
- Complete failure to detect the target object.

We observe that TTT partially mitigates these errors, though some failures persist, leaving room for further improvement.

For future work, we plan to incorporate temporal cues to better capture object dynamics and further reduce the failure patterns identified above.

References

[1] Yuqian Fu, Runze Wang, Yanwei Fu, Danda Pani Paudel, Xuanjing Huang, and Luc Van Gool. Objectrelator: Enabling cross-view object relation understanding in ego-centric and exo-centric videos. *ICCV*, 2025. 1

[2] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *CVPR*, 2024. 1

[3] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. 1

[4] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. 1

[5] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023. 1

[6] Zheng Zhang, Yeyao Ma, Enming Zhang, and Xiang Bai. Psalm: Pixelwise segmentation with large multi-modal model. In *ECCV*, 2024. 1

[7] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *NeurIPS*, 2023. 1

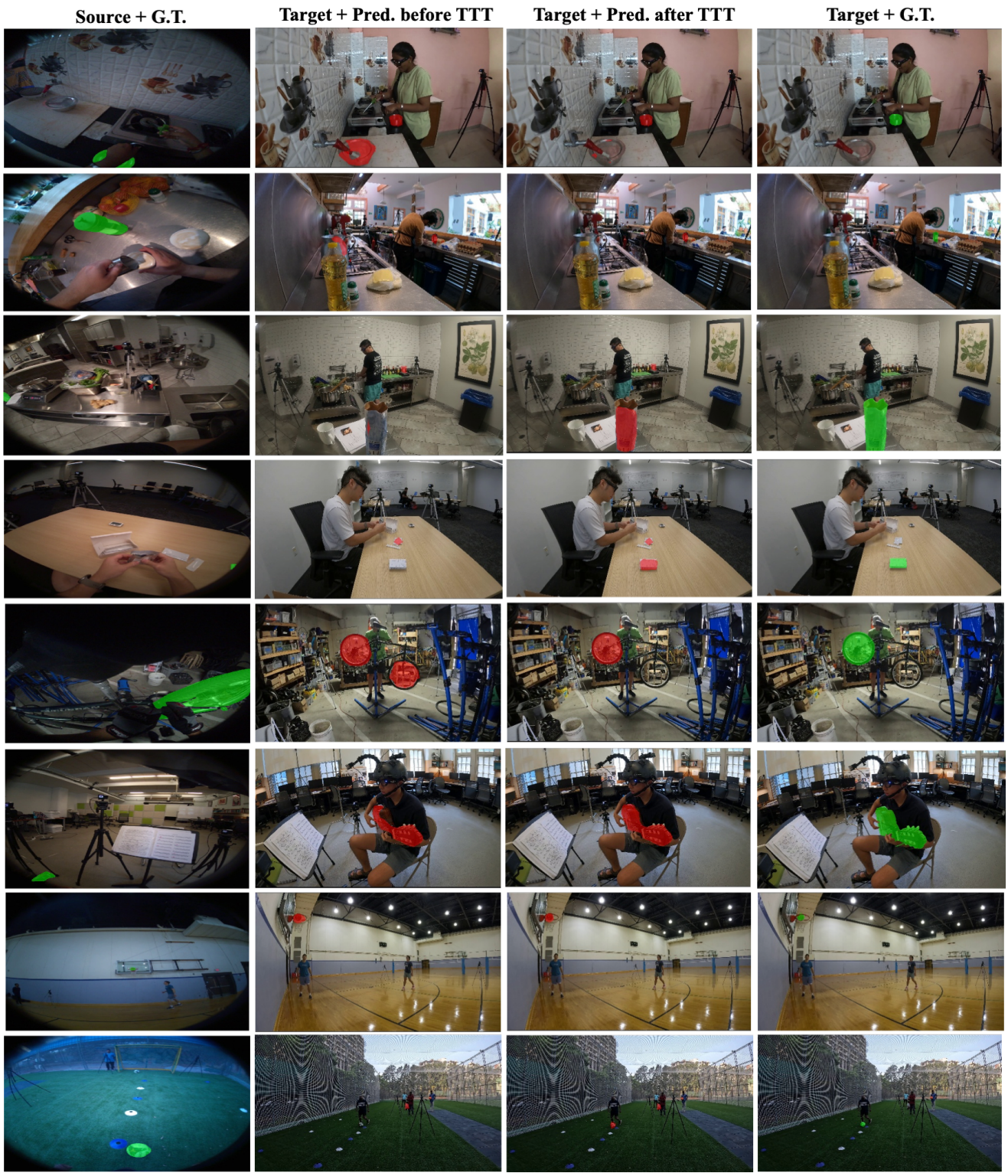

Figure 2. **More qualitative results on the Ego-Exo4D correspondence benchmark (Ego2Exo).** Each row shows a representative sample. Rows 1–3 correspond to cooking, row 4 to health, row 5 to bike repair, row 6 to music performance, row 7 to basketball, and row 8 to soccer.

Figure 3. **More qualitative results on the Ego-Exo4D correspondence benchmark (Exo2Ego).** Each row shows a representative sample. Rows 1–3 correspond to cooking, row 4 to health, row 5 to bike repair, row 6 to music performance, row 7 to basketball, and row 8 to soccer.

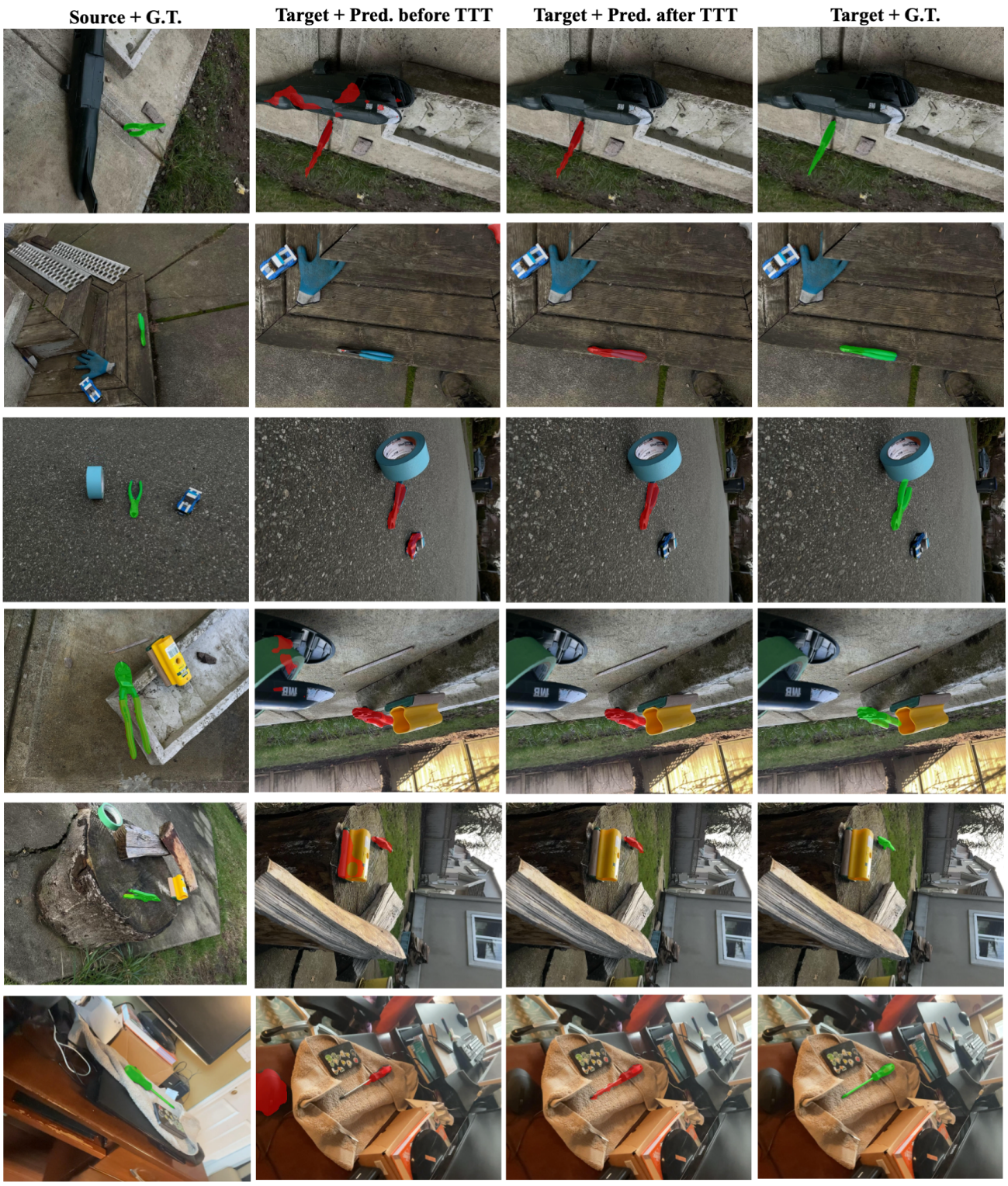

Figure 4. **More qualitative results on the HANDAL-X benchmark.** Each row shows a representative sample.