Supplementary Material
Calibrating Panoramic Depth Estimation for Practical Localization and Mapping

Junho Kim[1], Eun Sun Lee[1], and Young Min Kim[1, 2]

[1] Dept. of Electrical and Computer Engineering, Seoul National University

[2] Interdisciplinary Program in Artificial Intelligence and INMC, Seoul National University

{82magnolia, eunsunlee, youngmin.kim}@snu.ac.kr

A. Implementation Details

A.1. Loss Functions for Test-Time Training

As explained in Section 3.1, our calibration method involves fine-tuning the depth estimation network using three training objectives. In this section we explain how each objectives are implemented, along with the detailed hyperparameter setups.

Stretch Loss The goal of stretch loss is to mitigate the domain gap that occurs from depth scale changes in small or large scenes as shown in Figure B.2. The loss minimizes the difference between the stretched depth values against the original depth prediction, namely

$$\mathcal{L}_{\text{S}}=\begin{cases}\sum_{k\in\mathcal{K}_s}\|\hat{D}-\mathcal{S}_{\text{dpt}}^{1/k}(F_\Theta(\mathcal{S}_{\text{img}}^k(I)))\|_2 & \text{if avg}(\hat{D})<\delta_1\\ \sum_{k\in\mathcal{K}_l}\|\hat{D}-\mathcal{S}_{\text{dpt}}^{1/k}(F_\Theta(\mathcal{S}_{\text{img}}^k(I)))\|_2 & \text{if avg}(\hat{D})>\delta_2\\ 0 & \text{otherwise,}\end{cases} \tag{1}$$

where $\text{avg}(\hat{D})$ is the pixel-wise average for the depth map $\hat{D}=F_\Theta(I)$, and $\mathcal{K}_l=\{\sigma,\sigma^2\},\mathcal{K}_s=\{1/\sigma,1/\sigma^2\}$ are the stretch factors used for contracting and enlarging panoramas. We use the publicly available codebase from Sun et al. [23] to implement the stretching operations $\mathcal{S}_{\text{img}}^k,\mathcal{S}_{\text{dpt}}^k$, and set $\delta_1{=}1,\delta_2{=}2.5,\sigma{=}0.8$.

Chamfer and Normal Loss Along with stretch loss that enforces scale consistency, Chamfer loss and Normal loss impose fine-grained geometric consistency. Both losses operate by creating synthetic views rendered at random translations and rotations, where we adapt the codebase of Zioulis et al. [26] to implement the rendering operation. First, given a panorama image I, Chamfer loss is given as follows,

$$\mathcal{L}_{\text{C}}=\sum_{\mathbf{x}\in\mathcal{B}(\hat{D})}\min_{\mathbf{y}\in\mathcal{B}(D_{\text{warp}})}\|\tilde{R}\mathbf{x}+\tilde{t}-\mathbf{y}\|_2^2, \tag{2}$$

where $\mathcal{B}(\hat{D})$ is the point cloud created from the original depth prediction and $\mathcal{B}(D_{\text{warp}})=\mathcal{B}(F_\Theta(\mathcal{W}(I,\hat{D};\tilde{R},\tilde{t})))$ is the point cloud from the depth prediction made at a synthesized view from a randomly chosen pose $\tilde{R},\tilde{t}$ near the origin. We implement the Chamfer loss using the `chamfer_distance` function from the PyTorch3D library [19].

The Normal loss imbues an additional level of geometric consistency by aligning the normal vectors between the predictions for the original and synthetic views. The Normal loss is defined as follows,

$$\mathcal{L}_{\text{N}}=\sum_{\mathbf{x}\in\mathcal{B}(\hat{D})}(\tilde{R}\mathcal{N}(\mathbf{x})\cdot(\tilde{R}\mathbf{x}+\tilde{t}-\operatorname*{argmin}_{\mathbf{y}\in\mathcal{B}(D_{\text{warp}})}\|\tilde{R}\mathbf{x}+\tilde{t}-\mathbf{y}\|_2))^2 \tag{3}$$

where D_{warp} is a depth map from an arbitrary translation and rotation $\tilde{R},\tilde{t}$, and $\mathcal{N}(\mathbf{x})$ is the normal vector at point $\mathbf{x}$. We implement the Normal loss using the `estimate_pointcloud_normals` function from PyTorch3D [19] and set the number of ball queries as 15.

A.2. Robot Navigation

We implement the navigation agent similar to Active Neural SLAM [7] and use the Habitat simulator [21] for evaluating the application of our calibration method on robot navigation. As explained in Section 4.1, we consider an agent that receives a panorama image and noisy odometry sensor reading as inputs and draws an occupancy grid map. While the original implementation of Active Neural SLAM [7] trains the entire set of navigation modules end-to-end, we use the pre-trained depth estimation network F_Θ from Albanis et al. [2] and only train the policy network P_Ψ and pose estimator network C_Φ on the Gibson training split [21]. For depth calibration, the augmentation factor is $N_{\text{aug}}=10$ in all our experiments. The test-time training process caches images as shown in Figure 3, using the first $N_{\text{fwd}}=25$ images for the exploration and SLAM tasks and

$N_{\text{fwd}} = 3$ images for the point goal navigation task. We use a larger number of cached images for exploration and SLAM tasks as the tasks generally take longer steps compared to the point goal task.

A.3. Map-Free Localization

We implement a structure-based localization method [20] based on the setup explained in Section 4.2, The query image I_q is localized against a small 3D map $\mathcal{B}(\hat{D}_{\text{ref}})$ created from a reference image I_{ref}. To elaborate, we first generate synthetic panoramas at $N_t \times N_r$ poses and global/local features. During localization, the features are compared against the query image features to choose candidate poses and refine them using PnP-RANSAC [16, 10]. In our implementation, we use NetVLAD [3] for global features and SuperPoint [9] for local features, which are both widely used for visual localization [5, 20]. Further, we set the number of translations as $N_t = 100$ and rotations as $N_r = 8$. For rotations, we assume that the gravity direction is known and generate N_r rotation matrices by only varying the yaw angle values.

B. Additional Experimental Details and Results

B.1. Baseline Comparison in Depth Estimation

In Section 5.1, we establish comparisons against various baselines for depth estimation amidst domain changes. For evaluation we use the Stanford 2D-3D-S [4] and OmniScenes [13] datasets. Note for OmniScenes we use the 'turtlebot' split as other splits contain moving human hands and bodies whose ground-truth depth values are not available. Below we elaborate on the domain and baseline setups, and provide the additional experimental results of depth estimation.

B.1.1 Domain Setup

Online Adaptation For online adaptation, we evaluate our method in 10 domain shifts using the Stanford 2D-3D-S [4] and OmniScenes [13] datasets. The domain shifts shown in Figure 5 are implemented as follows:

- **Dataset Shift:** We do not apply any additional transformations to the images. The images from the tested datasets are directly used for evaluation.
- **Low Lighting:** We lower each pixel intensity by 25%.
- **White Balance:** We apply the following transformation matrix to the raw RGB color values: $\begin{pmatrix} 0.7 & 0 & 0 \\ 0 & 0.9 & 0 \\ 0 & 0 & 0.8 \end{pmatrix}$.
- **Gamma:** We set the image gamma to 1.5.
- **Speckle:** We use the `random_noise` function from the scikit-image library, where we set the speckle noise variance parameter to 0.06.
- **Gaussian:** We use the same library as in speckle noise, where we set the Gaussian noise variance parameter to 0.005.
- **Salt and Pepper:** We use the same library as in speckle noise, where we randomly perturb 0.5% of the image pixels.
- **Large Scene:** For OmniScenes [13], we select the `wedding`, `lounge`, and `lobby` scenes and for Stanford 2D-3D-S [4] we select all rooms labelled as `hallway` and `auditorium`.
- **Small Scene:** For OmniScenes [13], we select the `bride_room`, `makeup_room`, and `pyebaek` scenes and for Stanford 2D-3D-S [4] we select all rooms labelled as `pantry`, `WC`, `storage_room` and `copy_room`.
- **Rotations:** We apply a random rotation on the test images with yaw angles sampled from $\mathcal{U}(-\pi, \pi)$, roll angles sampled from $\mathcal{U}(-\pi/8, \pi/8)$, and pitch angles sampled from $\mathcal{U}(-\pi/8, \pi/8)$.

Offline Adaptation For offline adaptation, we separately evaluate depth estimation in each room for OmniScenes [13] and Stanford 2D-3D-S [4]. Specifically, we select 5% or 10% of panorama images in each room for training, and evaluate using the remaining images. Note that for the Stanford 2D-3D-S dataset, many rooms contain less than 20 panoramas, which means that often only a *single* image is used for adaptation. To cope with data scarcity, we apply data augmentation from Section 3.2 for $N_{\text{aug}} = 20$ times in the 5% case and $N_{\text{aug}} = 10$ times in the 10% case to increase the test-time training data.

B.1.2 Baseline Setup

For evaluating our calibration method, we test against seven baselines in the main paper. Here we elaborate on the implementations of each baseline, along with three additional baselines which we make detailed comparisons in Section B.1.3. In the offline setup, all baselines are trained for a single epoch to ensure fair comparison.

Tent and Batch Normalization Statistics Update Introduced by Wang et al. [24], Tent proposes to only train the affine parameters from the batch normalization layer during test-time training. Similar to Tent, Schneider et al. [22] propose to only update the batch normalization statistics during

adaptation. We adapt both baselines to our setup while for Tent we modify the original entropy-based training objective to our training objective in Equation 1 to accommodate for the task change from classification to depth prediction.

Flip Consistency Originally developed for self-supervised visual odometry [17], flip consistency imposes consistency against the flipped input image. Formally, the baseline minimizes the flip consistency loss given as follows,

$$\mathcal{L}_{\text{flip}} = \|F_\Theta(\mathcal{T}_{\text{flip}}(I)) - \mathcal{T}_{\text{flip}}(F_\Theta(I))\|_2, \quad (4)$$

where $\mathcal{T}_{\text{flip}}(\cdot)$ is the horizontal flipping operation.

Mask Consistency Inspired from Mate [18], the mask consistency baseline operates by first creating a randomly masked image and imposing depth consistency against the original prediction. Let $\tilde{M} \in \mathbb{R}^{H \times W}$ denote the random mask generated for each test sample. Then, the baseline is trained with the following objective,

$$\mathcal{L}_{\text{mask}} = \|\tilde{M} \circ F_\Theta(I) - \tilde{M} \circ F_\Theta(\tilde{M} \circ I)\|_2, \quad (5)$$

where $\circ$ is the member-wise product operation. We implement the random masking operation by first splitting the input panorama into $N_h \times N_w$ square patches and randomly discarding 10% of the patches, where we set $N_h = 4, N_w = 8$.

Photometric Consistency Similar to the loss functions often used for self-supervised depth estimation [11, 14], the photometric consistency baseline imposes consistency between the synthesized view using depth estimation results and the original view. The baseline first generates a synthetic panorama located at translation $\tilde{t}$ and rotation $\tilde{R}$ from the origin, namely $I_{\text{warp}} = \mathcal{W}(I, \hat{D}; \tilde{R}, \tilde{t})$. Then, the baseline minimizes the following loss,

$$\mathcal{L}_{\text{photo}} = \|I - \mathcal{W}(I_{\text{warp}}, D_{\text{warp}}; \tilde{R}^{-1}, \tilde{t}^{-1})\|_2, \quad (6)$$

where D_{warp} is the depth estimation using I_{warp}.

Pseudo Labelling First introduced by Lee et al. [15], the pseudo labelling baseline creates a pseudo ground-truth by aggregating depth predictions made at various rotated panoramas. Formally, the baseline minimizes the following objective,

$$\mathcal{L} = \|F_\Theta(I) - \frac{1}{K}\sum_{k=1}^{K} \mathcal{T}_{\text{rot}}(F_\Theta(\mathcal{T}_{\text{rot}}(I, \frac{2\pi}{K})), -\frac{2\pi}{K}), \|_2 \quad (7)$$

where $\mathcal{T}_{\text{rot}}(I, 2\pi/K)$ denotes the horizontal rotation of the input panorama I by $2\pi/K$ rad. In all our experiments we set $K = 4$.

Unsupervised Domain Adaptation We consider three unsupervised domain adaptation baselines: vanilla T^{2}Net, CrDoCo [8], and feature consistency [1]. All three baselines use the original source domain dataset during adaptation, which is the Matterport3D [6] dataset in our implementation. For each test sample, we randomly sample an image and ground-truth depth pair $(\tilde{I}_{\text{src}}, \tilde{D}_{\text{src}})$ and use them for adapting the network to the new domain. In addition, the baselines utilize a style transfer network $F_{\text{style}}(I, I_{\text{ref}})$ that transforms the input panorama I to match the style of the reference panorama I_{ref}. We implement F_{style} based on AdaIN [12], which is a widely used method for style transfer.

First, vanilla T^{2}Net imposes consistency between the style transferred depth prediction and the original depth prediction, namely

$$\mathcal{L}_{\text{T}^2\text{Net}} = \|F_\Theta(F_{\text{style}}(I_{\text{src}}, I)) - D_{\text{src}}\|_2. \quad (8)$$

Note that here the source domain image I_{src} is transformed to match the style of the target domain image I. While the original T^{2}Net imposes an additional set of adversarial losses based on GANs [25], we omit those losses and hence the baseline is named *vanilla* T^{2}Net. As our setup does not target a single transition from sim-to-real but a wide range of domain shifts and the number of test data is highly limited, it is infeasible to train a set of generators and discriminators for each domain shift.

CrDoCo [8] builds upon vanilla T^{2}Net and imposes an additional cross-domain consistency loss, namely

$$\mathcal{L}_{\text{CrDoCo}} = \mathcal{L}_{\text{T}^2\text{Net}} + \|F_\Theta(F_{\text{style}}(I, I_{\text{src}})) - F_\Theta(I)\|_2. \quad (9)$$

Conceptually, the new loss of CrDoCo transforms the target domain image to match the source domain image. It enforces the original depth prediction $F_\Theta(I)$ to follow the transformed prediction.

Finally, the feature consistency baseline [1] imposes an additional loss to impose consistency between the intermediate activations of the depth prediction network. This could be expressed as follows,

$$\begin{aligned}\mathcal{L}_{\text{feat}} = \mathcal{L}_{\text{CrDoCo}} &+ \|F^{\text{inter}}_\Theta(F_{\text{style}}(I, I_{\text{src}})) - F^{\text{inter}}_\Theta(I)\|_2 \\ &+ \|F^{\text{inter}}_\Theta(F_{\text{style}}(I_{\text{src}}, I)) - F^{\text{inter}}_\Theta(I_{\text{src}})\|_2, \end{aligned} \quad (10)$$

where F^{inter}_Θ is the intermediate layer activations of the depth estimation network F_Θ.

Ground-Truth Training To measure the upper-bound performance, we finally consider a baseline that uses the ground-truth data from the target domain. Specifically, the ground-truth training baseline minimizes the following loss,

$$\mathcal{L}_{\text{gt}} = \|F_\Theta(I) - D_{\text{gt}}\|_2, \quad (11)$$

where D_{gt} is the ground-truth depth map for image I.

Method	Area (m^2)	Collisions
No Adaptation	28.4169	7822
Flip Consistency	28.8958	**6440**
Mask Consistency	28.8796	7481
Pseudo Labelling	27.9863	7324
Ours	**30.5147**	6589

(a) Exploration

Method	PSNR
No Adaptation	7.2667
Flip Consistency	7.4840
Mask Consistency	7.5850
Pseudo Labelling	7.5000
Ours	**7.8404**

(b) SLAM w/ Fixed Trajectory

Table B.1: Additional metrics for the exploration and SLAM task in robot navigation.

B.1.3 Full Experimental Results

We report the full experimental results for depth estimation, where we present results from i) Stanford 2D-3D-S [4], ii) OmniScenes [13], and iii) the aggregated total results. Here we compare our calibration method against the baselines using six metrics, namely mean absolute error (MAE), absolute relative difference (Abs. Rel.), squared relative difference (Sq. Rel.), root mean squared error (RMSE), log root mean squared error (RMSE (Log)), and inlier ratio. Each metric is defined as follows:

- **MAE:** $\sum_{u,v} \frac{|D[u,v]-D_{\text{gt}}[u,v]|}{H*W}$
- **Abs. Rel.:** $\sum_{u,v} \frac{|D[u,v]-D_{\text{gt}}[u,v]|}{H*W*D_{\text{gt}}[u,v]}$
- **Sq. Rel.:** $\sum_{u,v} \frac{|D[u,v]-D_{\text{gt}}[u,v]|^2}{H*W*D_{\text{gt}}[u,v]}$
- **RMSE:** $\sqrt{\sum_{u,v} \frac{|D[u,v]-D_{\text{gt}}[u,v]|^2}{H*W}}$
- **RMSE (Log):** $\sqrt{\sum_{u,v} \frac{|\log D[u,v]-\log D_{\text{gt}}[u,v]|^2}{H*W}}$
- **Inlier Ratio:** $\sum_{u,v} \frac{1}{H*W} \mathbb{1}\{\max(\frac{D[u,v]}{D_{\text{gt}}[u,v]}, \frac{D_{\text{gt}}[u,v]}{D[u,v]}) < \lambda\}$, where λ is a pre-defined inlier threshold.

As shown in Table B.4 to Table B.39, our method outperforms most of the baselines in all tested metrics. We also display visualizations of the depth values before and after adaptation in Figure B.2, where our adaptation scheme can largely alleviate the quality deterioration from domain shifts. The depth estimation results suggest that our calibration method can serve as an effective enhancement scheme in practical depth estimation scenarios.

B.1.4 Loss Function Comparison

We additionally establish comparisons between the loss functions used in our calibration method (normal, stretch, Chamfer) against the loss functions used in the baselines. To this end we run a small experiment where we evaluate offline adaptation on two rooms in OmniScenes [13] (Room 5 and Wedding Hall) using 10% of the available images for training and the rest for testing. Here Room 5 exemplifies the 'dataset shift' case, whereas Wedding Hall exemplifies the 'large scene' case explained from Section B.1.1. We additionally apply an augmentation for each loss function by $N_{\text{aug}} = 10$ and measure the mean absolute error on the test samples. As shown in Table B.2, both normal loss and Chamfer loss outperform the other loss functions in Room 5 while stretch loss shows large improvements in the wedding hall scene. The fine-grained geometric consistencies from normal and Chamfer loss, along with the scale consistencies imposed from stretch loss enable our calibration method to function in a wide variety of depth estimation scenarios.

Loss Function	Room 5	Wedding Hall
No Adaptation	0.4839	1.1704
Flip	0.4605	1.2746
Mask	0.3644	1.3302
Photometric	0.4889	1.1177
Pseudo Labelling	0.4435	1.2935
Normal	**0.2699**	1.3437
Stretch	0.4741	**0.9167**
Chamfer	0.2917	1.3537

Table B.2: Mean absolute error of various test-time training loss functions measured from rooms in OmniScenes [13].

B.2. Robot Navigation

Experimental Setup In all our experiments, we set the maximum number of steps per episode to 500, while the point goal navigation task terminates whenever the agent stops within 0.2m of its estimated goal position. Also, note that the Chamfer distance metric shown in Table 2b is measured by treating the occupied regions in the grid map as a 2D point cloud. For the SLAM task under fixed trajectory shown in Table 2b, we collect the trajectories by having the 'No Adaptation' robot agent to explore in each episode for 500 steps.

Additional Results We report additional metrics and visualizations for the exploration and SLAM tasks in Table B.1 and Figure B.1. First, we show the average explored area and the total number of collisions occurred during exploration. Our method achieves the highest explored area while exhibiting a lower collision count than most of the competing methods. In addition, we display the PSNR between the estimated grid maps and the ground truth, where our method attains the highest grid map similarity. This is further verified through the qualitative samples in Figure B.1, where the grid map generated using our calibration scheme best aligns with the ground-truth. Therefore, our method could effectively function in various robot navigation tasks to enhance their performance in challenging deployment scenarios.

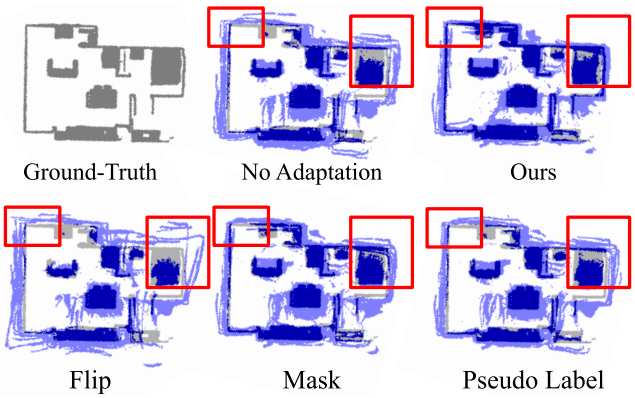
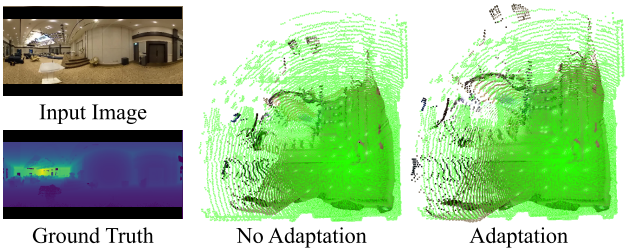
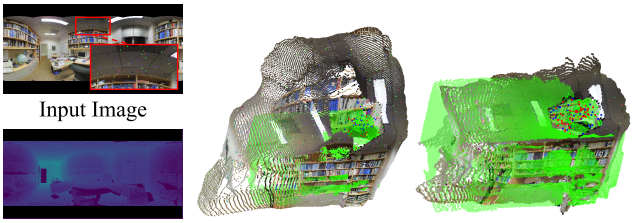


Figure B.1: Qualitative result of grid maps from navigation task. We display the ground-truth map (grey) and the estimated grid map (blue) from the same sequence of actions.



(a) Depth Adaptation Results in Large Scenes



(b) Depth Adaptation Results in Salt and Pepper Noise

Figure B.2: Qualitative visualization of depth estimation before and after adaptation. We overlay the ground-truth depth values in green.

B.3. Map-Free Localization

As explained in Section 5.3, we evaluate map-free localization by querying 10 images that are captured within a distance threshold $\delta = 2$m of each reference image. Also, for test-time training we augment the data by $N_{\text{aug}} = 20$ and cope with data scarcity. In this section we additionally report results for $\delta = 1, 3$m. Table B.3 shows the localization results at various distance thresholds, where our method outperforms the baselines in most tested setups. Along with robot navigation, our method demonstrates large amounts of performance enhancements in map-free localization, where the refined geometry of the depth maps play a crucial role for accurate localization.

Method	t-error (m)	R-error (°)	Accuracy (0.1m,5°)	Accuracy (0.2m,10°)
No Adatation	0.09	**0.54**	0.54	0.96
Flip	0.09	0.58	0.57	0.98
Mask	0.07	0.58	0.66	0.98
CrDoCo	0.07	0.61	0.70	**0.99**
SSL	0.08	0.61	0.58	0.95
Ours	**0.05**	0.60	**0.83**	0.96

(a) Query images within 1m of reference image

Method	t-error (m)	R-error (°)	Accuracy (0.1m,5°)	Accuracy (0.2m,10°)
No Adatation	0.26	1.35	0.20	0.42
Flip	0.22	1.26	0.24	0.46
Mask	0.19	1.43	0.29	0.51
CrDoCo	0.19	1.41	0.31	0.53
SSL	0.23	1.38	0.24	0.46
Ours	**0.15**	**1.13**	**0.38**	**0.60**

(b) Query images within 3m of reference image

Table B.3: Additional results of map-free visual localization compared against the baselines in OmniScenes [13]. Note that the translation and rotation error thresholds for calculating accuracy is denoted as $(d\,\text{m}, \theta°)$.

.

References

[1] Hiroyasu Akada, S. Bhat, Ibraheem Alhashim, and Peter Wonka. Self-supervised learning of domain invariant features for depth estimation. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 997–1007, 2021.

[2] Georgios Albanis, Nikolaos Zioulis, Petros Drakoulis, V. Gkitsas, Vladimiros Sterzentsenko, Federico Álvarez, Dimitrios Zarpalas, and Petros Daras. Pano3d: A holistic benchmark and a solid baseline for 360° depth estimation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3722–3732, 2021.

[3] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[4] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

[5] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Áron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part I*, pages 690–708. Springer, 2022.

[6] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017.

[7] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning

to explore using active neural slam. *arXiv preprint arXiv:2004.05155*, 2020.

[8] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1791–1800, 2019.

[9] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR Deep Learning for Visual SLAM Workshop*, 2018.

[10] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[11] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency, 2017.

[12] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017.

[13] Junho Kim, Changwoon Choi, Hojun Jang, and Young Min Kim. Piccolo: Point cloud-centric omnidirectional localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3313–3323, October 2021.

[14] Marvin Klingner, Jan-Aike Termöhlen, Jonas Mikolajczyk, and Tim Fingscheidt. Self-Supervised Monocular Depth Estimation: Solving the Dynamic Object Problem by Semantic Guidance. In *European Conference on Computer Vision (ECCV)*, 2020.

[15] Dong-Hyun Lee. Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013.

[16] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal Of Computer Vision*, 81:155–166, 2009.

[17] Bin Li, Mu Hu, Shuling Wang, Lianghao Wang, and Xiaojin Gong. Self-supervised visual-lidar odometry with flip consistency. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3844–3852, 2021.

[18] Muhammad Jehanzeb Mirza, Inkyu Shin, Wei Lin, Andreas Schriebl, Kunyang Sun, Jaesung Choe, Horst Possegger, Mateusz Koziński, In-So Kweon, Kun-Jin Yoon, and Horst Bischof. Mate: Masked autoencoders are online 3d test-time learners. *ArXiv*, abs/2211.11432, 2022.

[19] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020.

[20] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, 2019.

[21] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[22] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *ArXiv*, abs/2006.16971, 2020.

[23] Hao-Wen Ting, Cheng Sun, and Hwann-Tzong Chen. Self-supervised 360° room layout estimation. *ArXiv*, abs/2203.16057, 2022.

[24] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

[25] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *European Conference on Computer Vision*, 2018.

[26] Nikolaos Zioulis, Antonis Karakottas, Dimitrios Zarpalas, Federico Álvarez, and Petros Daras. Spherical view synthesis for self-supervised 360° depth estimation. *2019 International Conference on 3D Vision (3DV)*, pages 690–699, 2019.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4339	0.1934	0.1418	0.6019	0.0953	0.7367	0.9618	0.9861
Schneider et al. [22]	0.5302	0.2057	0.2127	0.815	0.1221	0.6679	0.9089	0.9579
Tent [24]	0.4278	0.1906	0.1388	0.5949	0.0943	0.7468	0.9624	0.9863
Flip Consistency	0.4203	0.1863	0.1341	0.5941	0.0917	0.7462	0.9619	0.9864
Mask Consistency	0.3976	0.1745	0.1232	0.5687	0.0883	0.7879	0.962	0.9866
Photometric Consistency	0.434	0.1958	0.1404	0.6027	0.0955	0.7235	0.9643	0.987
Pseudo Labelling	0.4087	0.1819	0.1281	0.5743	0.0909	0.7617	0.9637	0.9873
Vanilla T^{2}Net [25]	0.4182	0.1781	0.1334	0.6171	0.0921	0.7786	0.9547	0.9839
CrDoCo [8]	0.4099	0.1797	0.126	0.5821	0.0907	0.7741	0.9603	0.9864
Feature Consistency	0.4152	0.1768	0.1306	0.6118	0.0915	0.7821	0.9551	0.984
Ground-Truth Training	0.3161	0.1411	0.0899	0.4893	0.0764	0.8621	0.9758	0.991
Ours	0.3245	0.1448	0.0928	0.4756	0.0779	0.8686	0.9714	0.989

Table B.4: Offline adaptation using 5% of the panorama images for training in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4345	0.1951	0.1430	0.6029	0.0953	0.7335	0.9626	0.9866
Schneider et al. [22]	0.5317	0.2072	0.2144	0.8169	0.1220	0.6627	0.9087	0.9585
Tent [24]	0.4280	0.192	0.1396	0.5957	0.0942	0.7455	0.9634	0.9868
Flip Consistency	0.4044	0.1808	0.1263	0.5743	0.0897	0.7631	0.9633	0.9872
Mask Consistency	0.4022	0.1775	0.1252	0.5732	0.0893	0.7834	0.9612	0.9867
Photometric Consistency	0.4253	0.1926	0.1367	0.5845	0.0933	0.7364	0.9659	0.9879
Pseudo Labelling	0.4167	0.1846	0.1311	0.5907	0.0916	0.7580	0.9630	0.9873
Vanilla T^{2}Net [25]	0.4210	0.1806	0.1344	0.6210	0.0927	0.7738	0.9536	0.9841
CrDoCo [8]	0.4112	0.1814	0.1274	0.5850	0.0911	0.7719	0.9598	0.9865
Feature Consistency	0.4192	0.1775	0.1335	0.6233	0.0925	0.7794	0.9531	0.9836
Ground-Truth Training	0.2981	0.1363	0.0813	0.4543	0.0731	0.8754	0.9793	0.9923
Ours	0.3192	0.1434	0.0907	0.4674	0.0767	0.8732	0.9725	0.9896

Table B.5: Offline adaptation using 10% of the panorama images for training in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4334	0.1946	0.1431	0.6030	0.0957	0.7362	0.9616	0.9863
Schneider et al. [22]	0.4053	0.1555	0.1368	0.7470	0.1028	0.8033	0.9487	0.9794
Tent [24]	0.4198	0.1848	0.1355	0.6320	0.0955	0.7727	0.9608	0.9859
Flip Consistency	0.4700	0.1903	0.1628	0.8111	0.1059	0.7318	0.9372	0.9742
Mask Consistency	0.4898	0.2069	0.1705	0.8097	0.1095	0.6713	0.9324	0.9731
Photometric Consistency	0.4498	0.2083	0.1479	0.6656	0.1023	0.6615	0.9555	0.9862
Pseudo Labelling	0.4533	0.2043	0.1493	0.7110	0.1035	0.6910	0.9464	0.9801
Vanilla T^{2}Net [25]	0.4025	0.1767	0.1259	0.6345	0.0936	0.7909	0.9563	0.9831
CrDoCo [8]	0.4989	0.2115	0.1779	0.8306	0.1117	0.6668	0.9311	0.9704
Feature Consistency	0.6594	0.2402	0.3025	1.1390	0.1546	0.5887	0.8267	0.9172
Ground-Truth Training	0.2243	0.0922	0.0622	0.4245	0.0619	0.9143	0.9801	0.9930
Ours	0.3382	0.1433	0.0989	0.5393	0.0816	0.8432	0.9667	0.9902

Table B.6: Online adaptation in dataset shift evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.3766	0.2117	0.1378	0.4868	0.0966	0.7060	0.9715	0.9893
Schneider et al. [22]	0.3611	0.1944	0.1115	0.4837	0.0894	0.7520	0.9797	0.9930
Tent [24]	0.3495	0.1977	0.1215	0.4616	0.0938	0.7643	0.9740	0.9901
Flip Consistency	0.2823	0.1727	0.0833	0.3662	0.0850	0.8193	0.9782	0.9925
Mask Consistency	0.3134	0.1999	0.0948	0.3870	0.0940	0.7231	0.9710	0.9918
Photometric Consistency	0.3486	0.2066	0.1201	0.4373	0.0971	0.7083	0.9714	0.9895
Pseudo Labelling	0.3498	0.2166	0.1127	0.4254	0.0983	0.6855	0.9692	0.9910
Vanilla T^{2}Net [25]	0.3286	0.2126	0.1058	0.3930	0.0975	0.7163	0.9690	0.9911
CrDoCo [8]	0.3268	0.2063	0.0998	0.4068	0.0964	0.7131	0.9682	0.9916
Feature Consistency	0.4863	0.2433	0.2117	0.7122	0.1478	0.5800	0.8350	0.9256
Ground-Truth Training	0.1409	0.0835	0.0355	0.2284	0.0548	0.9411	0.9863	0.9948
Ours	0.1971	0.1099	0.0498	0.3042	0.0650	0.9132	0.9848	0.9942

Table B.7: Online adaptation in low lighting evaluated in the OmniScenes [13] dataset..

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.3929	0.2215	0.1412	0.4980	0.1000	0.6707	0.9686	0.9882
Schneider et al. [22]	0.3638	0.1952	0.1113	0.4854	0.0895	0.7459	0.9805	0.9934
Tent [24]	0.3656	0.2072	0.1250	0.4733	0.0974	0.7315	0.9711	0.9891
Flip Consistency	0.2879	0.1727	0.0790	0.3757	0.0858	0.8123	0.9768	0.9919
Mask Consistency	0.3261	0.2037	0.0976	0.4063	0.0958	0.7104	0.9696	0.9913
Photometric Consistency	0.3948	0.2368	0.1348	0.4736	0.1054	0.5850	0.9678	0.9895
Pseudo Labelling	0.3400	0.2120	0.1041	0.4097	0.0973	0.7001	0.9695	0.9911
Vanilla T^2Net [25]	0.3381	0.2128	0.1056	0.4033	0.0968	0.7219	0.9720	0.9915
CrDoCo [8]	0.3301	0.2085	0.0999	0.4066	0.0968	0.7103	0.9690	0.9914
Feature Consistency	0.4502	0.2350	0.1839	0.6507	0.1348	0.6048	0.8644	0.9497
Ground-Truth Training	0.1445	0.0850	0.0355	0.2338	0.0560	0.9359	0.9855	0.9944
Ours	0.2107	0.1154	0.0519	0.3228	0.0679	0.9006	0.9833	0.9936

Table B.8: Online adaptation in white balance change evaluated in the OmniScenes [13] dataset..

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.3782	0.2116	0.1269	0.4778	0.0946	0.7262	0.9754	0.9910
Schneider et al. [22]	0.3615	0.1941	0.1104	0.4831	0.0891	0.7509	0.9810	0.9934
Tent [24]	0.3474	0.1957	0.1108	0.4477	0.0911	0.7835	0.9776	0.9917
Flip Consistency	0.3026	0.1886	0.0861	0.3696	0.0883	0.7824	0.9791	0.9930
Mask Consistency	0.3281	0.2068	0.0958	0.3980	0.0952	0.6991	0.9736	0.9927
Photometric Consistency	0.3681	0.2283	0.1267	0.4361	0.1012	0.6431	0.9734	0.9911
Pseudo Labelling	0.3248	0.2018	0.0946	0.3890	0.0920	0.7500	0.9768	0.9928
Vanilla T^2Net [25]	0.3031	0.1921	0.0883	0.3667	0.0896	0.7788	0.9782	0.9926
CrDoCo [8]	0.3105	0.1954	0.0886	0.3855	0.0915	0.7581	0.9751	0.9926
Feature Consistency	0.3747	0.1998	0.1324	0.5403	0.1115	0.6917	0.9159	0.9758
Ground-Truth Training	0.1354	0.0799	0.0325	0.2215	0.0531	0.9453	0.9875	0.9952
Ours	0.1990	0.1080	0.0474	0.3112	0.0648	0.9142	0.9851	0.9942

Table B.9: Online adaptation in image gamma change evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5290	0.1680	0.1696	0.8141	0.0975	0.7440	0.9399	0.9789
Schneider et al. [22]	0.7638	0.2101	0.3221	1.3617	0.1702	0.5857	0.8289	0.9272
Tent [24]	0.5366	0.1678	0.1726	0.8776	0.1014	0.7461	0.9370	0.9780
Flip Consistency	0.6313	0.1721	0.2220	1.0901	0.1132	0.7247	0.9035	0.9601
Mask Consistency	0.6255	0.1745	0.2131	1.0730	0.1093	0.7273	0.9085	0.9662
Photometric Consistency	0.5512	0.1768	0.1750	0.8885	0.1013	0.6995	0.9405	0.9815
Pseudo Labelling	0.5747	0.1737	0.1857	0.9581	0.1025	0.7324	0.9261	0.9764
Vanilla T^2Net [25]	0.6003	0.1687	0.2001	1.0290	0.1061	0.7483	0.9162	0.9690
CrDoCo [8]	0.6517	0.1825	0.2295	1.1306	0.1132	0.7116	0.9027	0.9617
Feature Consistency	0.8238	0.2163	0.3541	1.4397	0.1496	0.6082	0.8331	0.9258
Ground-Truth Training	0.3663	0.1147	0.1058	0.6516	0.0739	0.8773	0.9708	0.9898
Ours	0.4790	0.1494	0.1435	0.7819	0.0911	0.8029	0.9510	0.9855

Table B.10: Online adaptation in large scenes evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.2921	0.1454	0.0738	0.4064	0.0750	0.8674	0.9787	0.9931
Schneider et al. [22]	0.2907	0.1388	0.0691	0.4183	0.0718	0.8625	0.9874	0.9954
Tent [24]	0.2862	0.1424	0.0710	0.4029	0.0752	0.8730	0.9794	0.9934
Flip Consistency	0.2898	0.1499	0.0699	0.3888	0.0758	0.8549	0.9832	0.9946
Mask Consistency	0.2459	0.1317	0.0540	0.3321	0.0693	0.8851	0.9863	0.9954
Photometric Consistency	0.2560	0.1302	0.0587	0.3583	0.0702	0.8889	0.9831	0.9943
Pseudo Labelling	0.2732	0.1407	0.0624	0.3699	0.0716	0.8783	0.9857	0.9953
Vanilla T^2Net [25]	0.2453	0.1328	0.0542	0.3300	0.0698	0.8819	0.9860	0.9953
CrDoCo [8]	0.2353	0.1269	0.0503	0.3216	0.0676	0.8905	0.9863	0.9953
Feature Consistency	0.2665	0.1283	0.0606	0.3835	0.0763	0.8547	0.9759	0.9904
Ground-Truth Training	0.1809	0.0930	0.0343	0.2679	0.0562	0.9325	0.9877	0.9954
Ours	0.2121	0.1058	0.0428	0.3119	0.0624	0.9139	0.9857	0.9946

Table B.11: Online adaptation in small scenes evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4585	0.2720	0.2294	0.6489	0.1443	0.5798	0.8498	0.9426
Schneider et al. [22]	0.4867	0.2878	0.2346	0.6636	0.1491	0.5088	0.8310	0.9432
Tent [24]	0.4419	0.2570	0.2021	0.6299	0.1413	0.5924	0.8583	0.9469
Flip Consistency	0.4128	0.2389	0.1609	0.6048	0.1270	0.6184	0.8907	0.9652
Mask Consistency	0.4368	0.2358	0.1746	0.6737	0.1352	0.6165	0.8678	0.9526
Photometric Consistency	0.4316	0.2574	0.1929	0.6076	0.1338	0.5979	0.8788	0.9593
Pseudo Labelling	0.4421	0.2413	0.1786	0.6687	0.1364	0.6013	0.8667	0.9536
Vanilla T^2Net [25]	0.4044	0.2565	0.1891	0.5642	0.1298	0.6314	0.8880	0.9613
CrDoCo [8]	0.4214	0.2265	0.1668	0.6600	0.1314	0.6457	0.8741	0.9555
Feature Consistency	0.6067	0.2913	0.2990	0.9284	0.2001	0.4699	0.7231	0.8552
Ground-Truth Training	0.2604	0.1583	0.0864	0.3917	0.0916	0.7987	0.9488	0.9833
Ours	0.4089	0.2106	0.1486	0.6182	0.1324	0.6070	0.8686	0.9593

Table B.12: Online adaptation in camera rotations evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5029	0.2770	0.3114	0.7194	0.1768	0.5950	0.8021	0.8799
Schneider et al. [22]	0.3800	0.2071	0.1240	0.5075	0.0953	0.7070	0.9723	0.9914
Tent [24]	0.4718	0.2585	0.2693	0.7245	0.1812	0.6205	0.8183	0.8916
Flip Consistency	0.3416	0.2064	0.1161	0.4584	0.1090	0.7121	0.9401	0.9753
Mask Consistency	0.3818	0.2004	0.1404	0.5899	0.1245	0.7092	0.9015	0.9578
Photometric Consistency	0.4544	0.2597	0.2660	0.6843	0.1672	0.6366	0.8445	0.9097
Pseudo Labelling	0.3274	0.1854	0.1078	0.4754	0.1070	0.7623	0.9339	0.9721
Vanilla T^2Net [25]	0.3324	0.2043	0.1369	0.4648	0.1126	0.7495	0.9298	0.9681
CrDoCo [8]	0.3383	0.1929	0.1102	0.4906	0.1051	0.7404	0.9408	0.9781
Feature Consistency	0.4046	0.2115	0.1539	0.6142	0.1302	0.6660	0.8876	0.9531
Ground-Truth Training	0.1868	0.1096	0.0550	0.2947	0.0720	0.8907	0.9674	0.9860
Ours	0.2557	0.1349	0.0717	0.4037	0.0851	0.8445	0.9597	0.9837

Table B.13: Online adaptation in gaussian noise evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.7473	0.4189	0.4784	0.9412	0.1649	0.3244	0.7849	0.9368
Schneider et al. [22]	0.3805	0.2064	0.1217	0.5048	0.0948	0.7144	0.9723	0.9923
Tent [24]	0.6900	0.3872	0.4067	0.8798	0.1576	0.3752	0.8195	0.9481
Flip Consistency	0.4892	0.3025	0.2027	0.5750	0.1281	0.4454	0.9114	0.9793
Mask Consistency	0.4715	0.2995	0.1914	0.5531	0.1281	0.4464	0.9129	0.9799
Photometric Consistency	0.7913	0.4655	0.4944	0.9309	0.1776	0.1959	0.7345	0.9340
Pseudo Labelling	0.4980	0.3064	0.2038	0.5793	0.1286	0.4273	0.9164	0.9803
Vanilla T^2Net [25]	0.5771	0.3668	0.3141	0.6923	0.1511	0.4214	0.8409	0.9531
CrDoCo [8]	0.4024	0.2534	0.1463	0.4898	0.1130	0.5797	0.9392	0.9852
Feature Consistency	0.4755	0.2377	0.1979	0.7094	0.1417	0.5922	0.8486	0.9376
Ground-Truth Training	0.2030	0.1171	0.0575	0.3034	0.0677	0.8789	0.9764	0.9925
Ours	0.3311	0.1876	0.1080	0.4455	0.0930	0.7160	0.9620	0.9905

Table B.14: Online adaptation in salt and pepper noise evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5285	0.2970	0.3467	0.7498	0.1787	0.5729	0.8000	0.8826
Schneider et al. [22]	0.3918	0.2119	0.1344	0.5313	0.0981	0.7048	0.9646	0.9890
Tent [24]	0.4945	0.2753	0.2902	0.7334	0.1783	0.5988	0.8170	0.8958
Flip Consistency	0.3779	0.2323	0.1408	0.4878	0.1154	0.6493	0.9347	0.9751
Mask Consistency	0.3641	0.2137	0.1273	0.5021	0.1117	0.6906	0.9327	0.9764
Photometric Consistency	0.5642	0.3368	0.4919	0.8553	0.2010	0.5685	0.7775	0.8630
Pseudo Labelling	0.3761	0.2233	0.1510	0.5260	0.1182	0.6849	0.9232	0.9695
Vanilla T^2Net [25]	0.3691	0.2299	0.1603	0.4953	0.1169	0.6954	0.9246	0.9698
CrDoCo [8]	0.3581	0.2093	0.1233	0.5006	0.1096	0.7097	0.9355	0.9771
Feature Consistency	0.4086	0.2218	0.1542	0.5971	0.1280	0.6413	0.8958	0.9611
Ground-Truth Training	0.2036	0.1194	0.0602	0.3150	0.0744	0.8745	0.9683	0.9878
Ours	0.2753	0.1509	0.0830	0.4160	0.0881	0.8233	0.9603	0.9850

Table B.15: Online adaptation in speckle noise evaluated in the OmniScenes [13] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4656	0.2533	0.2171	0.6551	0.1215	0.63	0.9077	0.9704
Schneider et al. [22]	0.5102	0.2684	0.2686	0.7714	0.1308	0.5776	0.8871	0.9632
Tent [24]	0.4645	0.2527	0.2163	0.6537	0.1213	0.6314	0.908	0.9705
Flip Consistency	0.4586	0.2494	0.2085	0.6441	0.1207	0.6344	0.9091	0.9706
Mask Consistency	0.444	0.2426	0.1967	0.6271	0.118	0.6522	0.914	0.9722
Photometric Consistency	0.4674	0.2559	0.2188	0.6538	0.1218	0.6254	0.9071	0.9706
Pseudo Labelling	0.4476	0.2471	0.2006	0.6214	0.1179	0.6411	0.9133	0.9732
Vanilla T^2Net [25]	0.4404	0.2402	0.1944	0.6268	0.1183	0.6565	0.9136	0.9716
CrDoCo [8]	0.444	0.2437	0.1971	0.6257	0.1181	0.6497	0.9134	0.9722
Feature Consistency	0.4374	0.2365	0.19	0.6223	0.1172	0.6611	0.9155	0.9722
Ground-Truth Training	0.4023	0.2234	0.1678	0.5646	0.1108	0.6903	0.9252	0.9762
Ours	0.4313	0.2365	0.1896	0.6057	0.1152	0.6652	0.9185	0.9744

Table B.16: Offline adaptation using 5% of the panorama images for training in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4601	0.2491	0.2136	0.6560	0.1217	0.6387	0.9091	0.9698
Schneider et al. [22]	0.4973	0.2618	0.2557	0.7548	0.1276	0.5865	0.8957	0.9662
Tent [24]	0.4592	0.2486	0.2129	0.6549	0.1215	0.6398	0.9094	0.9699
Flip Consistency	0.4513	0.2461	0.2051	0.6383	0.1202	0.6435	0.9115	0.9706
Mask Consistency	0.4461	0.2411	0.2004	0.6419	0.1197	0.6554	0.9124	0.9702
Photometric Consistency	0.4623	0.2510	0.2152	0.6573	0.1220	0.6352	0.9085	0.9698
Pseudo Labelling	0.4505	0.2448	0.2038	0.6431	0.1206	0.6452	0.9108	0.9701
Vanilla T^2Net [25]	0.4430	0.2385	0.1983	0.6419	0.1198	0.6610	0.9120	0.9694
CrDoCo [8]	0.4444	0.2419	0.1991	0.6362	0.1191	0.6537	0.9131	0.9711
Feature Consistency	0.4415	0.2378	0.1971	0.6385	0.1196	0.6618	0.9116	0.9693
Ground-Truth Training	0.4073	0.2255	0.1741	0.5789	0.1126	0.6881	0.9236	0.9749
Ours	0.4313	0.2350	0.1914	0.6134	0.1168	0.6686	0.9164	0.9722

Table B.17: Offline adaptation using 10% of the panorama images for training in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4538	0.2461	0.2069	0.6504	0.1208	0.6444	0.9113	0.9708
Schneider et al. [22]	0.4381	0.2351	0.1957	0.6793	0.1156	0.6446	0.9266	0.9780
Tent [24]	0.4397	0.2361	0.1949	0.6680	0.1209	0.6683	0.9159	0.9718
Flip Consistency	0.3951	0.2253	0.1538	0.6183	0.1145	0.6785	0.9221	0.9750
Mask Consistency	0.4120	0.2333	0.1638	0.6594	0.1186	0.6552	0.9152	0.9709
Photometric Consistency	0.4752	0.2792	0.2150	0.6596	0.1277	0.5670	0.9030	0.9718
Pseudo Labelling	0.4038	0.2304	0.1608	0.6329	0.1165	0.6684	0.9186	0.9729
Vanilla T^2Net [25]	0.4017	0.2336	0.1683	0.6236	0.1166	0.6762	0.9156	0.9726
CrDoCo [8]	0.4091	0.2313	0.1644	0.6626	0.1175	0.6654	0.9137	0.9714
Feature Consistency	0.6116	0.2687	0.3168	1.0522	0.1906	0.5360	0.7803	0.8794
Ground-Truth Training	0.2624	0.1415	0.0921	0.4524	0.0866	0.8366	0.9585	0.9860
Ours	0.2994	0.1594	0.1136	0.5008	0.0944	0.8086	0.9511	0.9824

Table B.18: Online adaptation in dataset shift evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4018	0.2295	0.1645	0.5567	0.1157	0.6505	0.9194	0.9743
Schneider et al. [22]	0.4826	0.2759	0.2379	0.6949	0.1263	0.5577	0.9031	0.9725
Tent [24]	0.3910	0.2225	0.1565	0.5652	0.1157	0.6682	0.9220	0.9749
Flip Consistency	0.3391	0.2086	0.1213	0.4775	0.1091	0.7039	0.9281	0.9774
Mask Consistency	0.3427	0.2140	0.1234	0.4817	0.1100	0.6942	0.9258	0.9769
Photometric Consistency	0.3874	0.2238	0.1527	0.5597	0.1188	0.6626	0.9179	0.9720
Pseudo Labelling	0.3671	0.2306	0.1411	0.5012	0.1136	0.6662	0.9213	0.9765
Vanilla T^2Net [25]	0.3670	0.2360	0.1464	0.4988	0.1158	0.6636	0.9150	0.9735
CrDoCo [8]	0.3475	0.2109	0.1240	0.5055	0.1116	0.6933	0.9216	0.9751
Feature Consistency	0.4633	0.2501	0.2024	0.7093	0.1484	0.5870	0.8515	0.9350
Ground-Truth Training	0.2728	0.1594	0.0853	0.4119	0.0932	0.7954	0.9498	0.9851
Ours	0.2995	0.1712	0.1001	0.4517	0.0987	0.7696	0.9440	0.9815

Table B.19: Online adaptation in low lighting evaluated in the Stanford 2D-3D-S [4] dataset..

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4484	0.2503	0.1943	0.6187	0.1289	0.6043	0.8947	0.9617
Schneider et al. [22]	0.4819	0.2749	0.2358	0.6904	0.1259	0.5581	0.9036	0.9731
Tent [24]	0.4364	0.2426	0.1856	0.6328	0.1305	0.6231	0.8982	0.9622
Flip Consistency	0.3763	0.2190	0.1448	0.5630	0.1237	0.6775	0.9015	0.9626
Mask Consistency	0.3567	0.2250	0.1324	0.4933	0.1130	0.6819	0.9193	0.9755
Photometric Consistency	0.4583	0.2692	0.2000	0.6268	0.1314	0.5745	0.8938	0.9657
Pseudo Labelling	0.4116	0.2579	0.1670	0.5454	0.1225	0.6046	0.9069	0.9725
Vanilla T^2Net [25]	0.3934	0.2481	0.1597	0.5294	0.1200	0.6352	0.9087	0.9729
CrDoCo [8]	0.3506	0.2190	0.1276	0.4926	0.1119	0.6902	0.9202	0.9761
Feature Consistency	0.5548	0.2730	0.2651	0.8570	0.1847	0.5144	0.7729	0.8830
Ground-Truth Training	0.2680	0.1574	0.0842	0.4079	0.0921	0.8039	0.9501	0.9849
Ours	0.3208	0.1779	0.1121	0.4968	0.1084	0.7516	0.9292	0.9736

Table B.20: Online adaptation in white balance change evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4480	0.2630	0.2045	0.6001	0.1214	0.6096	0.9060	0.9729
Schneider et al. [22]	0.4787	0.2730	0.2278	0.6811	0.1247	0.5567	0.9056	0.9743
Tent [24]	0.4306	0.2524	0.1906	0.5968	0.1201	0.6343	0.9117	0.9745
Flip Consistency	0.3722	0.2373	0.1459	0.4994	0.1144	0.6608	0.9186	0.9767
Mask Consistency	0.3439	0.2207	0.1275	0.4743	0.1098	0.6949	0.9245	0.9777
Photometric Consistency	0.5021	0.3105	0.2447	0.6446	0.1357	0.5093	0.8789	0.9662
Pseudo Labelling	0.3923	0.2499	0.1577	0.5182	0.1178	0.6302	0.9145	0.9757
Vanilla T^2Net [25]	0.3835	0.2490	0.1601	0.5140	0.1187	0.6445	0.9090	0.9732
CrDoCo [8]	0.3575	0.2325	0.1391	0.4880	0.1137	0.6727	0.9164	0.9759
Feature Consistency	0.4370	0.2273	0.1832	0.6742	0.1415	0.6208	0.8598	0.9396
Ground-Truth Training	0.2600	0.1526	0.0836	0.4019	0.0900	0.8167	0.9527	0.9854
Ours	0.2887	0.1678	0.0993	0.4387	0.0956	0.7895	0.9468	0.9826

Table B.21: Online adaptation in image gamma change evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4477	0.2187	0.2015	0.6987	0.1164	0.7078	0.9259	0.9698
Schneider et al. [22]	0.3996	0.1865	0.1607	0.6796	0.1048	0.7645	0.9473	0.9806
Tent [24]	0.4541	0.2178	0.2011	0.7520	0.1204	0.7067	0.9253	0.9696
Flip Consistency	0.4345	0.2180	0.1738	0.7062	0.1110	0.7038	0.9382	0.9784
Mask Consistency	0.4330	0.2170	0.1749	0.7296	0.1110	0.7044	0.9361	0.9776
Photometric Consistency	0.4519	0.2170	0.1975	0.7504	0.1214	0.6990	0.9260	0.9686
Pseudo Labelling	0.4341	0.2070	0.1745	0.7638	0.1136	0.7193	0.9303	0.9734
Vanilla T^2Net [25]	0.4243	0.2060	0.1698	0.7484	0.1096	0.7282	0.9350	0.9763
CrDoCo [8]	0.4212	0.2017	0.1665	0.7592	0.1082	0.7390	0.9356	0.9767
Feature Consistency	0.5788	0.2286	0.2871	1.1030	0.1693	0.6357	0.8420	0.9113
Ground-Truth Training	0.3121	0.1561	0.1243	0.5452	0.0894	0.8356	0.9605	0.9852
Ours	0.3471	0.1727	0.1423	0.5803	0.0960	0.8036	0.9546	0.9829

Table B.22: Online adaptation in large scenes evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.3692	0.2702	0.1673	0.4963	0.1289	0.5536	0.8942	0.9700
Schneider et al. [22]	0.7259	0.5169	0.5504	0.9410	0.1914	0.1590	0.6832	0.9222
Tent [24]	0.3686	0.2684	0.1670	0.5025	0.1292	0.5605	0.8937	0.9693
Flip Consistency	0.3494	0.2575	0.1489	0.4742	0.1277	0.5780	0.8986	0.9681
Mask Consistency	0.3504	0.2614	0.1492	0.4681	0.1254	0.5726	0.9052	0.9717
Photometric Consistency	0.3900	0.2870	0.1836	0.5210	0.1318	0.5234	0.8856	0.9683
Pseudo Labelling	0.3788	0.2817	0.1737	0.5015	0.1298	0.5329	0.8933	0.9700
Vanilla T^2Net [25]	0.3598	0.2692	0.1584	0.4775	0.1271	0.5572	0.9006	0.9705
CrDoCo [8]	0.3441	0.2565	0.1442	0.4637	0.1269	0.5813	0.8997	0.9684
Feature Consistency	0.3518	0.2558	0.1492	0.4838	0.1312	0.5715	0.8899	0.9635
Ground-Truth Training	0.3224	0.2387	0.1324	0.4428	0.1204	0.6365	0.9131	0.9731
Ours	0.3380	0.2485	0.1400	0.4612	0.1240	0.6030	0.9069	0.9717

Table B.23: Online adaptation in small scenes evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.6338	0.4173	0.5105	0.8961	0.1883	0.4565	0.7385	0.8736
Schneider et al. [22]	0.6689	0.4128	0.5905	1.0118	0.1957	0.4283	0.7307	0.8763
Tent [24]	0.6140	0.4044	0.4817	0.8734	0.1863	0.4710	0.7482	0.8805
Flip Consistency	0.5165	0.3283	0.2912	0.7463	0.1654	0.5099	0.7931	0.9093
Mask Consistency	0.5241	0.3456	0.3090	0.7420	0.1666	0.5014	0.7883	0.9060
Photometric Consistency	0.7020	0.4857	0.6141	0.9436	0.2006	0.4037	0.6973	0.8547
Pseudo Labelling	0.5337	0.3482	0.3172	0.7543	0.1689	0.4933	0.7826	0.9045
Vanilla T^2Net [25]	0.5671	0.4031	0.4120	0.7711	0.1782	0.4838	0.7641	0.8876
CrDoCo [8]	0.4875	0.3064	0.2586	0.7175	0.1576	0.5402	0.8119	0.9189
Feature Consistency	0.6049	0.3223	0.3282	0.9144	0.2048	0.4404	0.7076	0.8517
Ground-Truth Training	0.4140	0.2637	0.2149	0.6145	0.1382	0.6188	0.8560	0.9437
Ours	0.4910	0.2778	0.2513	0.7496	0.1659	0.5379	0.7986	0.9162

Table B.24: Online adaptation in camera rotations evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5816	0.2848	0.3196	0.8809	0.2088	0.5158	0.7474	0.8531
Schneider et al. [22]	0.5049	0.2948	0.2788	0.7356	0.1354	0.5389	0.8807	0.9617
Tent [24]	0.5788	0.2808	0.3131	0.9297	0.2143	0.5220	0.7478	0.8540
Flip Consistency	0.5882	0.2802	0.3002	0.9428	0.2032	0.5052	0.7474	0.8617
Mask Consistency	0.7381	0.3376	0.4076	1.1113	0.2568	0.3664	0.6149	0.7674
Photometric Consistency	0.4515	0.2441	0.2110	0.7100	0.1573	0.6191	0.8474	0.9269
Pseudo Labelling	0.5162	0.2587	0.2430	0.8323	0.1755	0.5605	0.8016	0.9005
Vanilla T^2Net [25]	0.4390	0.2497	0.2102	0.6776	0.1487	0.6234	0.8554	0.9349
CrDoCo [8]	0.6525	0.3030	0.3396	1.0192	0.2214	0.4432	0.6924	0.8277
Feature Consistency	0.4787	0.2518	0.2171	0.7482	0.1605	0.5882	0.8276	0.9187
Ground-Truth Training	0.3183	0.1869	0.1206	0.4952	0.1109	0.7418	0.9176	0.9701
Ours	0.4091	0.2044	0.1651	0.6682	0.1417	0.6495	0.8719	0.9427

Table B.25: Online adaptation in gaussian noise evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.8215	0.4977	0.7425	1.1088	0.1973	0.3717	0.6989	0.8608
Schneider et al. [22]	0.5000	0.2864	0.2689	0.7428	0.1341	0.5560	0.8826	0.9641
Tent [24]	0.7739	0.4674	0.6664	1.0653	0.1916	0.3970	0.7241	0.8759
Flip Consistency	0.4477	0.2872	0.2146	0.6100	0.1353	0.5787	0.8684	0.9546
Mask Consistency	0.4869	0.3141	0.2554	0.6591	0.1445	0.5340	0.8458	0.9437
Photometric Consistency	0.9862	0.6152	1.0352	1.2827	0.2288	0.2971	0.6131	0.7955
Pseudo Labelling	0.4995	0.3226	0.2622	0.6586	0.1451	0.5136	0.8429	0.9453
Vanilla T^2Net [25]	0.6522	0.4365	0.4794	0.8433	0.1797	0.4222	0.7406	0.8938
CrDoCo [8]	0.4263	0.2697	0.1957	0.6018	0.1323	0.6111	0.8732	0.9557
Feature Consistency	0.5416	0.2721	0.2674	0.8473	0.1814	0.5372	0.7844	0.8905
Ground-Truth Training	0.3294	0.1957	0.1308	0.4940	0.1077	0.7434	0.9217	0.9741
Ours	0.3843	0.2290	0.1690	0.5610	0.1189	0.6806	0.9031	0.9670

Table B.26: Online adaptation in salt and pepper noise evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5291	0.2777	0.2937	0.7796	0.1783	0.5324	0.7877	0.8992
Schneider et al. [22]	0.5032	0.2953	0.2651	0.7229	0.1342	0.5266	0.8850	0.9642
Tent [24]	0.5250	0.2719	0.2807	0.8240	0.1824	0.5404	0.7918	0.9003
Flip Consistency	0.6716	0.3124	0.3487	1.0187	0.2264	0.3972	0.6693	0.8278
Mask Consistency	0.5444	0.2713	0.2576	0.8487	0.1760	0.5043	0.7904	0.9045
Photometric Consistency	0.4359	0.2632	0.2325	0.6427	0.1417	0.6097	0.8771	0.9494
Pseudo Labelling	0.4622	0.2503	0.2096	0.7200	0.1507	0.5853	0.8496	0.9368
Vanilla T^2Net [25]	0.4258	0.2532	0.2073	0.6372	0.1400	0.6183	0.8751	0.9496
CrDoCo [8]	0.5459	0.2699	0.2550	0.8597	0.1752	0.5037	0.7910	0.9050
Feature Consistency	0.4766	0.2497	0.2181	0.7474	0.1582	0.5828	0.8321	0.9230
Ground-Truth Training	0.3062	0.1829	0.1178	0.4704	0.1052	0.7588	0.9307	0.9756
Ours	0.3375	0.1886	0.1266	0.5322	0.1125	0.7215	0.9218	0.9716

Table B.27: Online adaptation in speckle noise evaluated in the Stanford 2D-3D-S [4] dataset.

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4460	0.2163	0.1705	0.6222	0.1053	0.6960	0.9412	0.9801
Schneider et al. [22]	0.5226	0.2296	0.2340	0.7984	0.1254	0.6334	0.9006	0.9599
Tent [24]	0.4418	0.2143	0.1684	0.6173	0.1046	0.7028	0.9416	0.9803
Flip Consistency	0.4349	0.2104	0.1625	0.6132	0.1028	0.7035	0.9417	0.9804
Mask Consistency	0.4153	0.2005	0.1513	0.5910	0.0996	0.7361	0.9437	0.9811
Photometric Consistency	0.4467	0.2187	0.1703	0.6222	0.1055	0.6861	0.9425	0.9807
Pseudo Labelling	0.4235	0.2068	0.1558	0.5923	0.1012	0.7157	0.9445	0.9819
Vanilla T^{2}Net [25]	0.4267	0.2018	0.1567	0.6208	0.1021	0.7320	0.9390	0.9792
CrDoCo [8]	0.4229	0.2041	0.1531	0.5987	0.1012	0.7266	0.9424	0.9810
Feature Consistency	0.4237	0.1996	0.1533	0.6158	0.1013	0.7359	0.9400	0.9795
Ground-Truth Training	0.3490	0.1725	0.1196	0.5180	0.0895	0.7965	0.9565	0.9854
Ours	0.3653	0.1798	0.1297	0.5253	0.0921	0.7910	0.9512	0.9834

Table B.28: Full aggregated results for offline adaptation using 5% of the panorama images for training

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4443	0.2157	0.1699	0.6232	0.1054	0.6973	0.9422	0.9802
Schneider et al. [22]	0.5186	0.2280	0.2302	0.7932	0.1241	0.6336	0.9037	0.9614
Tent [24]	0.4399	0.2136	0.1676	0.6183	0.1046	0.7052	0.9428	0.9803
Flip Consistency	0.4223	0.2057	0.1564	0.5987	0.1013	0.7174	0.9435	0.9809
Mask Consistency	0.4190	0.2018	0.1539	0.5994	0.1009	0.7345	0.9426	0.9804
Photometric Consistency	0.4394	0.2149	0.1667	0.6123	0.1043	0.6978	0.9440	0.9810
Pseudo Labelling	0.4296	0.2076	0.1588	0.6107	0.1027	0.7149	0.9431	0.9807
Vanilla T^{2}Net [25]	0.4294	0.2027	0.1588	0.6290	0.1030	0.7307	0.9377	0.9785
CrDoCo [8]	0.4239	0.2045	0.1548	0.6045	0.1018	0.7268	0.9420	0.9806
Feature Consistency	0.4277	0.2005	0.1578	0.6291	0.1028	0.7345	0.9373	0.9781
Ground-Truth Training	0.3398	0.1703	0.1167	0.5019	0.0882	0.8039	0.9580	0.9857
Ours	0.3620	0.1784	0.1291	0.5231	0.0920	0.7951	0.9511	0.9830

Table B.29: Full aggregated results for offline adaptation using 10% of the panorama images for training

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4412	0.2143	0.1675	0.6211	0.1053	0.7012	0.9424	0.9804
Schneider et al. [22]	0.4178	0.1859	0.1593	0.7212	0.1077	0.7427	0.9403	0.9789
Tent [24]	0.4274	0.2044	0.1582	0.6457	0.1052	0.7329	0.9437	0.9805
Flip Consistency	0.4414	0.2037	0.1594	0.7375	0.1092	0.7115	0.9314	0.9745
Mask Consistency	0.4601	0.2170	0.1679	0.7523	0.1130	0.6652	0.9258	0.9723
Photometric Consistency	0.4595	0.2354	0.1735	0.6633	0.1120	0.6254	0.9355	0.9807
Pseudo Labelling	0.4344	0.2143	0.1537	0.6812	0.1085	0.6824	0.9358	0.9774
Vanilla T^{2}Net [25]	0.4022	0.1984	0.1421	0.6303	0.1024	0.7471	0.9408	0.9791
CrDoCo [8]	0.4646	0.2191	0.1727	0.7665	0.1139	0.6663	0.9245	0.9708
Feature Consistency	0.6412	0.2511	0.3080	1.1059	0.1683	0.5686	0.8090	0.9028
Ground-Truth Training	0.2388	0.1110	0.0736	0.4351	0.0713	0.8846	0.9719	0.9903
Ours	0.3234	0.1494	0.1045	0.5246	0.0865	0.8300	0.9607	0.9872

Table B.30: Full aggregated results for online adaptation in dataset shift

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.3862	0.2185	0.1480	0.5135	0.1039	0.6848	0.9516	0.9836
Schneider et al. [22]	0.4075	0.2255	0.1597	0.5643	0.1035	0.6778	0.9505	0.9852
Tent [24]	0.3653	0.2072	0.1349	0.5011	0.1022	0.7276	0.9542	0.9843
Flip Consistency	0.3040	0.1864	0.0978	0.4087	0.0942	0.7753	0.9591	0.9867
Mask Consistency	0.3246	0.2053	0.1057	0.4231	0.1001	0.7121	0.9537	0.9861
Photometric Consistency	0.3634	0.2132	0.1325	0.4840	0.1054	0.6909	0.9510	0.9828
Pseudo Labelling	0.3564	0.2219	0.1235	0.4543	0.1041	0.6781	0.9509	0.9855
Vanilla T^{2}Net [25]	0.3433	0.2215	0.1213	0.4334	0.1045	0.6962	0.9484	0.9844
CrDoCo [8]	0.3347	0.2081	0.1090	0.4445	0.1022	0.7055	0.9504	0.9853
Feature Consistency	0.4775	0.2459	0.2082	0.7111	0.1480	0.5827	0.8413	0.9292
Ground-Truth Training	0.1912	0.1125	0.0545	0.2984	0.0695	0.8855	0.9724	0.9911
Ours	0.2362	0.1333	0.0690	0.3605	0.0779	0.8584	0.9692	0.9894

Table B.31: Full aggregated results for online adaptation in low lighting

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4141	0.2325	0.1615	0.5441	0.1110	0.6454	0.9404	0.9781
Schneider et al. [22]	0.4089	0.2256	0.1588	0.5636	0.1034	0.6742	0.9511	0.9857
Tent [24]	0.3926	0.2207	0.1481	0.5342	0.1100	0.6901	0.9433	0.9788
Flip Consistency	0.3216	0.1904	0.1041	0.4472	0.1003	0.7608	0.9481	0.9807
Mask Consistency	0.3378	0.2118	0.1109	0.4395	0.1024	0.6995	0.9504	0.9853
Photometric Consistency	0.4190	0.2492	0.1597	0.5321	0.1153	0.5810	0.9396	0.9804
Pseudo Labelling	0.3673	0.2295	0.1281	0.4615	0.1069	0.6636	0.9456	0.9840
Vanilla T^2Net [25]	0.3592	0.2263	0.1262	0.4514	0.1057	0.6888	0.9478	0.9844
CrDoCo [8]	0.3379	0.2125	0.1105	0.4394	0.1026	0.7026	0.9504	0.9856
Feature Consistency	0.4901	0.2495	0.2149	0.7294	0.1538	0.5703	0.8295	0.9242
Ground-Truth Training	0.1916	0.1126	0.0541	0.3003	0.0698	0.8855	0.9720	0.9908
Ours	0.2527	0.1393	0.0749	0.3892	0.0834	0.8437	0.9627	0.9860

Table B.32: Full aggregated results for online adaptation in white balance change

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4048	0.2312	0.1565	0.5245	0.1048	0.6817	0.9489	0.9841
Schneider et al. [22]	0.4062	0.2242	0.1552	0.5587	0.1027	0.6768	0.9522	0.9861
Tent [24]	0.3792	0.2173	0.1413	0.5046	0.1022	0.7266	0.9524	0.9851
Flip Consistency	0.3292	0.2072	0.1089	0.4191	0.0983	0.7360	0.9560	0.9868
Mask Consistency	0.3341	0.2121	0.1079	0.4271	0.1008	0.6975	0.9549	0.9870
Photometric Consistency	0.4192	0.2597	0.1717	0.5157	0.1144	0.5920	0.9373	0.9816
Pseudo Labelling	0.3506	0.2202	0.1187	0.4383	0.1018	0.7043	0.9530	0.9863
Vanilla T^2Net [25]	0.3338	0.2138	0.1157	0.4229	0.1007	0.7275	0.9518	0.9852
CrDoCo [8]	0.3284	0.2096	0.1079	0.4246	0.1000	0.7255	0.9527	0.9862
Feature Consistency	0.3985	0.2103	0.1518	0.5914	0.1230	0.6646	0.8945	0.9620
Ground-Truth Training	0.1830	0.1076	0.0520	0.2904	0.0672	0.8962	0.9742	0.9915
Ours	0.2332	0.1308	0.0672	0.3599	0.0766	0.8666	0.9705	0.9898

Table B.33: Full aggregated results for online adaptation in image gamma change

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.4950	0.1892	0.1829	0.7659	0.1054	0.7289	0.9341	0.9751
Schneider et al. [22]	0.6117	0.2002	0.2547	1.0768	0.1429	0.6604	0.8783	0.9495
Tent [24]	0.5021	0.1887	0.1845	0.8251	0.1093	0.7296	0.9321	0.9745
Flip Consistency	0.5491	0.1913	0.2019	0.9298	0.1123	0.7160	0.9180	0.9677
Mask Consistency	0.5451	0.1922	0.1971	0.9296	0.1100	0.7177	0.9200	0.9710
Photometric Consistency	0.5097	0.1936	0.1844	0.8308	0.1097	0.6993	0.9344	0.9761
Pseudo Labelling	0.5160	0.1876	0.1810	0.8770	0.1071	0.7269	0.9279	0.9751
Vanilla T^2Net [25]	0.5268	0.1843	0.1874	0.9118	0.1076	0.7399	0.9241	0.9720
CrDoCo [8]	0.5554	0.1905	0.2032	0.9755	0.1111	0.7230	0.9164	0.9680
Feature Consistency	0.7215	0.2214	0.3261	1.2991	0.1578	0.6197	0.8368	0.9197
Ground-Truth Training	0.3437	0.1320	0.1135	0.6072	0.0804	0.8599	0.9665	0.9879
Ours	0.4239	0.1591	0.1430	0.6977	0.0931	0.8032	0.9525	0.9844

Table B.34: Full aggregated results for online adaptation in large scenes

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.3116	0.1770	0.0975	0.4292	0.0887	0.7878	0.9573	0.9872
Schneider et al. [22]	0.4010	0.2347	0.1911	0.5508	0.1021	0.6841	0.9103	0.9768
Tent [24]	0.3071	0.1743	0.0953	0.4282	0.0889	0.7938	0.9577	0.9873
Flip Consistency	0.3049	0.1772	0.0899	0.4105	0.0890	0.7847	0.9618	0.9879
Mask Consistency	0.2724	0.1646	0.0781	0.3666	0.0835	0.8059	0.9657	0.9894
Photometric Consistency	0.2900	0.1700	0.0904	0.3995	0.0858	0.7962	0.9584	0.9877
Pseudo Labelling	0.3000	0.1764	0.0906	0.4033	0.0864	0.7907	0.9623	0.9889
Vanilla T^2Net [25]	0.2743	0.1674	0.0806	0.3674	0.0843	0.7996	0.9643	0.9890
CrDoCo [8]	0.2629	0.1598	0.0741	0.3576	0.0826	0.8121	0.9643	0.9885
Feature Consistency	0.2881	0.1606	0.0831	0.4089	0.0902	0.7829	0.9541	0.9836
Ground-Truth Training	0.2168	0.1299	0.0592	0.3122	0.0725	0.8575	0.9688	0.9897
Ours	0.2440	0.1420	0.0674	0.3498	0.0780	0.8351	0.9657	0.9888

Table B.35: Full aggregated results for online adaptation in small scenes

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5254	0.3275	0.3367	0.7433	0.1611	0.5327	0.8073	0.9163
Schneider et al. [22]	0.5562	0.3355	0.3704	0.7965	0.1669	0.4781	0.7927	0.9177
Tent [24]	0.5076	0.3133	0.3088	0.7228	0.1585	0.5461	0.8163	0.9216
Flip Consistency	0.4524	0.2730	0.2106	0.6588	0.1417	0.5770	0.8534	0.9439
Mask Consistency	0.4701	0.2777	0.2259	0.6998	0.1472	0.5726	0.8375	0.9348
Photometric Consistency	0.5348	0.3445	0.3537	0.7358	0.1593	0.5238	0.8095	0.9194
Pseudo Labelling	0.4771	0.2821	0.2315	0.7014	0.1488	0.5601	0.8346	0.9349
Vanilla T^{2}Net [25]	0.4665	0.3125	0.2742	0.6432	0.1483	0.5751	0.8407	0.9332
CrDoCo [8]	0.4466	0.2570	0.2018	0.6819	0.1414	0.6054	0.8504	0.9415
Feature Consistency	0.6060	0.3031	0.3101	0.9231	0.2019	0.4586	0.7172	0.8539
Ground-Truth Training	0.3190	0.1985	0.1354	0.4767	0.1094	0.7300	0.9134	0.9682
Ours	0.4402	0.2362	0.1878	0.6684	0.1452	0.5806	0.8419	0.9428

Table B.36: Full aggregated results for online adaptation in camera rotations

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5329	0.2800	0.3145	0.7810	0.1890	0.5648	0.7812	0.8697
Schneider et al. [22]	0.4277	0.2406	0.1831	0.5946	0.1106	0.6428	0.9373	0.9801
Tent [24]	0.5126	0.2670	0.2860	0.8028	0.1938	0.5829	0.7914	0.8772
Flip Consistency	0.4357	0.2346	0.1864	0.6433	0.1450	0.6331	0.8665	0.9319
Mask Consistency	0.5178	0.2528	0.2424	0.7889	0.1750	0.5784	0.7921	0.8851
Photometric Consistency	0.4533	0.2537	0.2450	0.6941	0.1634	0.6299	0.8456	0.9163
Pseudo Labelling	0.3995	0.2134	0.1594	0.6116	0.1331	0.6853	0.8834	0.9448
Vanilla T^{2}Net [25]	0.3731	0.2216	0.1649	0.5460	0.1264	0.7014	0.9014	0.9554
CrDoCo [8]	0.4582	0.2349	0.1978	0.6924	0.1495	0.6270	0.8460	0.9207
Feature Consistency	0.4329	0.2269	0.1780	0.6653	0.1418	0.6363	0.8647	0.9400
Ground-Truth Training	0.2370	0.1391	0.0800	0.3712	0.0868	0.8339	0.9484	0.9799
Ours	0.3143	0.1614	0.1074	0.5047	0.1067	0.7701	0.9262	0.9681

Table B.37: Full aggregated results for online adaptation in Gaussian noise

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.7756	0.4490	0.5792	1.0052	0.1773	0.3425	0.7521	0.9078
Schneider et al. [22]	0.4261	0.2369	0.1779	0.5956	0.1098	0.6539	0.9381	0.9815
Tent [24]	0.7220	0.4178	0.5058	0.9506	0.1706	0.3835	0.7831	0.9205
Flip Consistency	0.4734	0.2967	0.2072	0.5884	0.1308	0.4963	0.8950	0.9699
Mask Consistency	0.4774	0.3051	0.2158	0.5936	0.1344	0.4798	0.8873	0.9661
Photometric Consistency	0.8657	0.5226	0.7008	1.0652	0.1971	0.2345	0.6882	0.8811
Pseudo Labelling	0.4986	0.3126	0.2261	0.6096	0.1349	0.4602	0.8883	0.9669
Vanilla T^{2}Net [25]	0.6058	0.3934	0.3772	0.7499	0.1620	0.4217	0.8026	0.9305
CrDoCo [8]	0.4115	0.2596	0.1652	0.5325	0.1204	0.5917	0.9140	0.9739
Feature Consistency	0.5007	0.2508	0.2244	0.7620	0.1569	0.5712	0.8241	0.9196
Ground-Truth Training	0.2512	0.1471	0.0855	0.3762	0.0830	0.8272	0.9555	0.9855
Ours	0.3514	0.2034	0.1313	0.4896	0.1029	0.7025	0.9395	0.9815

Table B.38: Full aggregated results for online adaptation in salt and pepper noise

Method	MAE	Abs. Rel.	Sq. Rel.	RMSE	RMSE (Log)	Inlier Ratio ($\lambda = 1.25$)	Inlier Ratio ($\lambda = 1.25^2$)	Inlier Ratio ($\lambda = 1.25^3$)
No Adaptation	0.5287	0.2896	0.3265	0.7612	0.1785	0.5574	0.7953	0.8889
Schneider et al. [22]	0.4343	0.2437	0.1843	0.6044	0.1119	0.6368	0.9342	0.9795
Tent [24]	0.5061	0.2740	0.2866	0.7680	0.1799	0.5765	0.8074	0.8975
Flip Consistency	0.4900	0.2629	0.2202	0.6904	0.1578	0.5531	0.8334	0.9189
Mask Consistency	0.4329	0.2357	0.1770	0.6344	0.1362	0.6195	0.8784	0.9490
Photometric Consistency	0.5152	0.3087	0.3929	0.7742	0.1784	0.5842	0.8155	0.8960
Pseudo Labelling	0.4090	0.2336	0.1734	0.6000	0.1306	0.6469	0.8951	0.9570
Vanilla T^{2}Net [25]	0.3907	0.2388	0.1782	0.5495	0.1257	0.6660	0.9057	0.9621
CrDoCo [8]	0.4298	0.2324	0.1736	0.6377	0.1346	0.6311	0.8803	0.9496
Feature Consistency	0.4346	0.2324	0.1786	0.6545	0.1395	0.6190	0.8715	0.9466
Ground-Truth Training	0.2428	0.1436	0.0822	0.3743	0.0862	0.8303	0.9539	0.9831
Ours	0.2990	0.1653	0.0996	0.4604	0.0974	0.7844	0.9456	0.9799

Table B.39: Full aggregated results for online adaptation in speckle noise