Generalizing to the Open World: Deep Visual Odometry with Online Adaptation Supplementary Materials

1. Learning settings

The optical flow network RAFT [3] consists of three components: (1) shared image feature encoder, (2) 4D feature correlation cost volume (correlation layer), and (3) RNN-based update operator. In the pretraining process, we train all these componenets together. As for online adaptation process, we deem that the extracted features may vary a lot in different environments, while the operations on feature are less sensitive to the environment change. Therefore, we fix correlation layer and RNN-based update operator during online adaptation and only learns feature encoder. The iterations of update operator is set 10 for both pretraining and online learning.

We pretrain the FlowNet and DepthNet with an additional PoseNet according to the training training setting of Competitive Collaboration [2] except that the motion segmentation mask M is not learned but calculated according to Monodepth2 [1]. The training data is augmented with random scaling, cropping, color jittering and horizontal flips. First, we pretrain FlowNet and DepthNet + PoseNet separately to get basic model parameters. Then we iteratively train FlowNet, DepthNet and PoseNet while keeping the other network weights fixed. While for online adaptation, we discard the PoseNet and use Essential matrix + PnP to solve pose.

Self-supervised learning of single-view depth estimation favors image sequences with small rotations and sufficient translations. However, in contrast to outdoor driving scenes (Cityscapes and KITTI), indoor datasets (TUM and NYUv2) usually have large rotations and small translations. In order to alleviate this issue, we increase the image translation by downsampling the videos by extracting one frame from every 10 frames.

During online adaptation, our VO framework achieves 10-14 fps including network forward inference, pose calculation, depth refinement, loss computation and network updating by gradient descent. The speed may vary slightly due to different RANSAC iterations for convergence.

The insights of learning dense flow rather than sparse matches are twofold: 1) self-supervised learning of sparse matches is challenging since photometric loss, as the only data term, is incapable of learning discriminative matches. In contrast, the learned dense optical flow is assumed generally accurate if the photometric loss becomes small enough; 2) optical flow, depth and pose are correlated by well-defined geometric constraint (*i.e.* scene flow). During self-supervised learning, the depth and flow networks will mutually help each other learn better estimations.

2. Updating of depth parameters

In the Section 3.2-3.3 of the original paper, the MAP of inverse depth z can be approximated [4] by the product of a Gaussian distribution for z and a Beta distribution for inlier ratio ρ:

$$q(z, \rho | a_t, b_t, \mu_t, \sigma_t^2) := Beta(\rho | a_t, b_t) \mathcal{N}(z_t | \mu_t, \sigma_t^2), \quad (1)$$

The parameter updating rules of $a_t, b_t, \mu_t, \sigma_t^2$ are listed as follows. For every new estimation z_t, the MAP of parameters is solved iteratively:

$$\begin{aligned} q(z, \rho | a_t, b_t, \mu_t, \sigma_t^2) = & p(z_t | z, \rho) \times \\ & q(z, \rho | a_{t-1}, b_{t-1}, \mu_{t-1}, \sigma_{t-1}^2), \end{aligned} \quad (2)$$

where:

$$p(z_t | z, \rho) = \rho \mathcal{N}(z_t | z, \tau^2) + (1 - \rho) \mathcal{U}(z_t | z_{\text{min}}, z_{\text{max}}). \quad (3)$$

Therefore, Eq. 2 can be written as:

$$\begin{aligned} & q(z, \rho | a_t, b_t, \mu_t, \sigma_t^2) = \\ & \quad [\rho \mathcal{N}(z_t | z, \tau^2) + (1 - \rho) \mathcal{U}(z_t | z_{\text{min}}, z_{\text{max}})] \times \\ & \quad Beta(\rho | a_{t-1}, b_{t-1}) \mathcal{N}(z | \mu_{t-1}, \sigma_{t-1}^2). \end{aligned} \quad (4)$$

During calculation, the multiplication of two Gaussian distributions can be written as:

$$\begin{aligned} & \mathcal{N}(z_t | z, \tau^2) \mathcal{N}(z | \mu_{t-1}, \sigma_{t-1}^2) = \\ & \quad \mathcal{N}(z_t | \mu_{t-1}, \tau^2 + \sigma_{t-1}^2) \mathcal{N}(z | m, s^2) \end{aligned} \quad (5)$$

where:

$$\begin{aligned} \frac{1}{s^2} &= \frac{1}{\sigma_{t-1}^2} + \frac{1}{\tau^2} \\ m &= s^2 (\frac{\mu_{t-1}}{\sigma_{t-1}^2} + \frac{z_t}{\tau^2}) \end{aligned} \quad (6)$$

In addition, since:

$$\Gamma(a,b) = \frac{1}{\rho}\frac{a}{a+b}\Gamma(a+1,b) = \frac{1}{1-\rho}\frac{b}{a+b}\Gamma(a,b+1) \tag{7}$$

Combining Eq. 5,6,7, Eq. 4 can be written as:

$$\begin{aligned} &q(z,\rho|a_t,b_t,\mu_t,\sigma_t^2) = \\ &\quad C_1\mathcal{N}(z|m,s^2)Beta(\rho|a_{t-1},b_{t-1}) + \\ &\quad C_2\mathcal{N}(z|\mu_{t-1},\sigma_{t-1}^2)Beta(\rho|a_{t-1},b_{t-1}+1)) \end{aligned} \tag{8}$$

where:

$$\begin{aligned} C_1 &= \frac{a_{t-1}}{a_{t-1}+b_{t-1}}\mathcal{N}(z_t|\mu_{t-1},\tau^2+\sigma_{t-1}^2) \\ C_2 &= \frac{b_{t-1}}{a_{t-1}+b_{t-1}}\mathcal{U}(z_t|z_{\min},z_{\max}) \end{aligned} \tag{9}$$

For convenience, we normalize C_1, C_2:

$$\begin{aligned} C_1' &= \frac{C_1}{C_1+C_2} \\ C_2' &= \frac{C_2}{C_1+C_2} \end{aligned} \tag{10}$$

Therefore, the mean and variance of the inverse depth are updated by:

$$\mu_t = C_1' m + C_2'\mu_{t-1} \tag{11}$$

$$\sigma_t^2 = C_1'(s^2+m^2) + C_2'(\sigma_{t-1}^2+\mu_{t-1}^2) \tag{12}$$

In order to update a, b in Beta distribution, we introduce e and f for convenience:

$$\begin{aligned} e =& \frac{a_t(a_t+1)}{(a_t+b_t)(a_t+b_t+1)} \\ =& C_1'\frac{(a_t+1)(a_t+2)}{(a_t+b_t+1)(a_t+b_t+2)} + \\ & C_2'\frac{a_t(a_t+1)}{(a_t+b_t+1)(a_t+b_t+2)} \end{aligned} \tag{13}$$

$$\begin{aligned} f = \frac{a_t}{a_t+b_t} =& C_1'\frac{a_{t-1}+1}{a_{t-1}+b_{t-1}+1} + \\ & C_2'\frac{a_{t-1}}{a_{t-1}+b_{t-1}+1} \end{aligned} \tag{14}$$

Then a_t, b_t are updated as follows:

$$a_t = \frac{e-f}{f-\frac{e}{f}} \tag{15}$$

$$b_t = \frac{1-f}{f}a_t \tag{16}$$

During depth refinement, the depth of remaining pixels will not be refined. However, in many cases the selected patches will cover the majority of depth map in a very short time (*e.g.* the case shown in Figure 3 in the original paper takes only 4 time steps), indicating that the majority of the keyframe depth will be refined.

3. Scale alignment for triangulated points

The mid-point triangulation method may get negative depth due to the numerical issue. Therefore, we assess each correspondence pair with respect to the angle of camera rays and filter out the ones around epipoles. We also filter the triangulated points with negative depth or ones projected out of the image region.

We use the selected sparse depth points for scale alignment. The scale is solved by aligning the calculated sparse depth with the corresponding depth position in the keyframe. We use RANSAC to increase the robustness of scale estimation. If the inlier count is still not enough after maximum iterations, we calculate the scale by aligning the median with triangulated and keyframe depth.

4. Additional VO results on TUM

We show more qualitative trajectory results on TUM dataset in Fig. 1 and Fig. 2. All these methods are pretrained on outdoor KITTI dataset and directly tested on indoor TUM dataset. When confronted with large domain shift, existing methods tend to fail while our method still recovers reasonable trajectories. It should be noted that we do not use re-localization and loop closure as did in classic methods to boost VO performance.

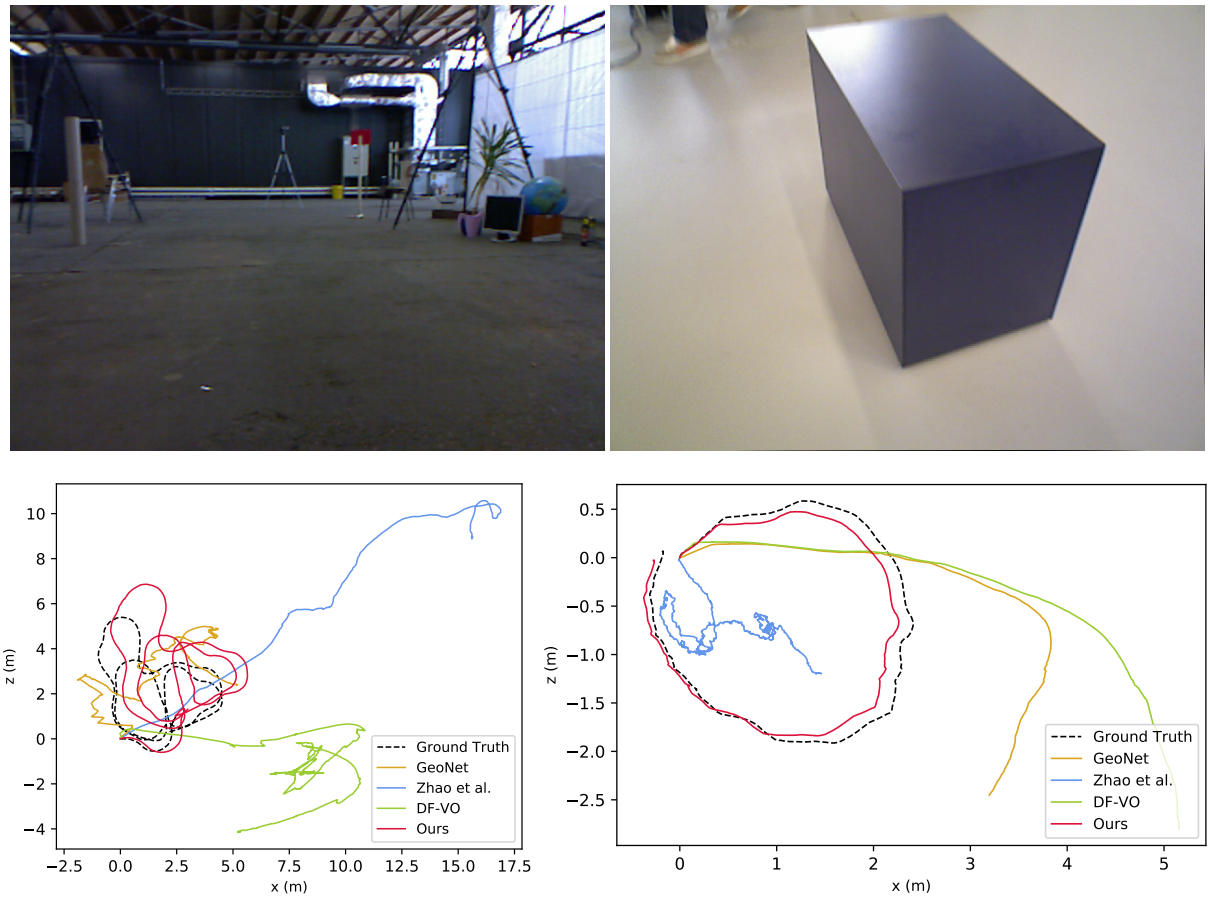
5. Depth results on NYUv2

We present single-view depth estimation results compared with Zhao *et al.* [6] and P^{2}Net [5]. The qualitative results on NYUv2 dataset are shown in Fig. 3. It can be seen that our method is able to recover more details and sharper edges than the other methods.

6. Qualitative results of ablation studies

We present qualitative results of ablation studies on various versions of our method. Fig. 4 shows the results that pretrained on KITTI and tested on TUM dataset. w/o RDS means without refined depth for online self-supervision; w/o RU means without predicted photometric uncertainty map. It can be seen that the trajectory error increases a lot if the refined depth is not used for self-supervision. In this case, the DepthNet is only learned by minimizing photometric loss, the convergence speed is much slower than supervising DepthNet with refined depth. Therefore, when facing large domain shift, the refined depth helps VO framework to maintain stable tracking and speeds up the online adaptation considerably.

Besides, it can be seen that the online learned photometric uncertainty (w/o PU) helps a lot on KITTI and TUM for pose estimation. This is because it downweights the regions with photometric inconstancy, occlusions and dynamic ob-



(c) fr2_pioneer_slam (d) fr3_cabinet

Figure 1. Additional visual odometry results on indoor TUM dataset.

jects, which helps recover more accurate pose by minimizing weighted photometric loss.

References

[1] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into Self-Supervised Monocular Depth Estimation. In *ICCV*, 2019. 1

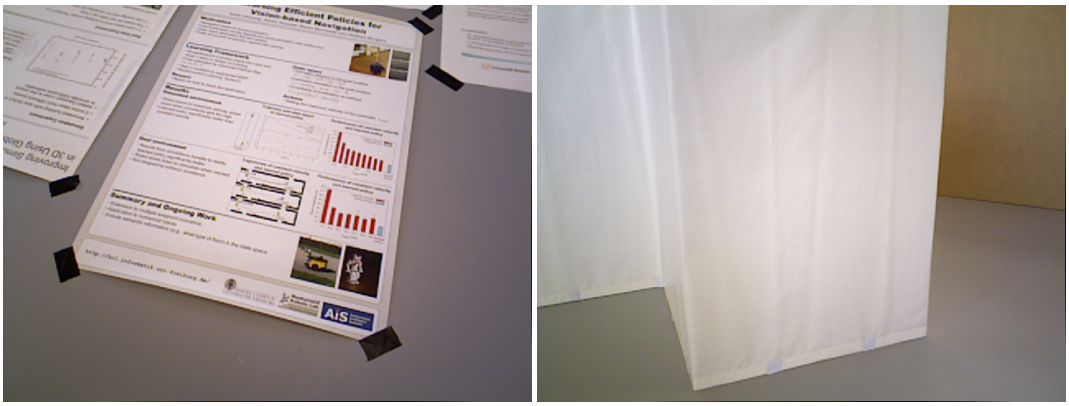
[2] Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J Black. Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation. In *CVPR*, 2019. 1

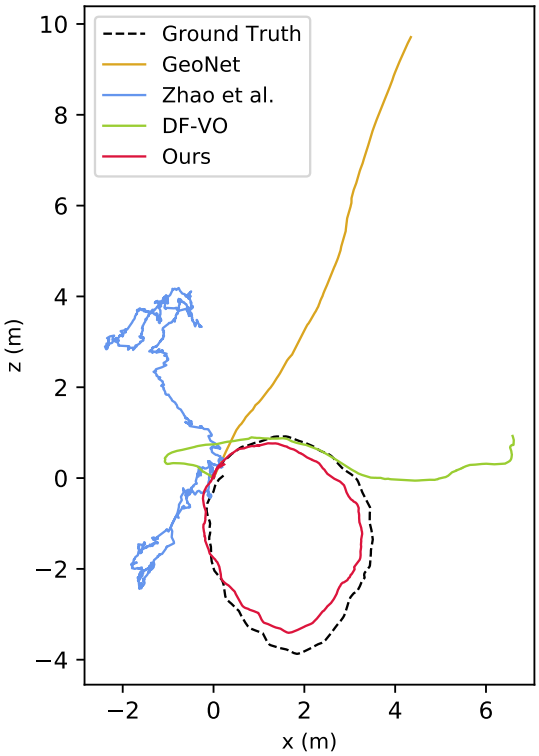
[3] Zachary Teed and Jia Deng. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In *ECCV*, 2020. 1

[4] George Vogiatzis and Carlos Hernández. Video-Based, Real-Time Multi-View Stereo. *Image and Vision Computing*, 29(7):434–441, 2011. 1

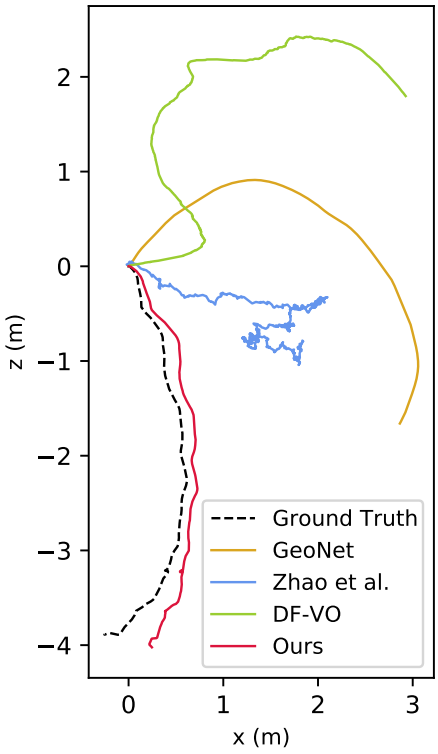
[5] Zehao Yu, Lei Jin, and Shenghua Gao. P2Net: Patch-match and Plane-regularization for Unsupervised Indoor Depth Estimation. In *ECCV*. 2

[6] Wang Zhao, Shaohui Liu, Yezhi Shu, and Yong-Jin Liu. Towards Better Generalization: Joint Depth-Pose Learning without PoseNet. In *CVPR*, 2020. 2





(c) fr3_nostr_texture_near_loop



(d) fr3_str_notexture_far

Figure 2. Additional visual odometry results on indoor TUM dataset.

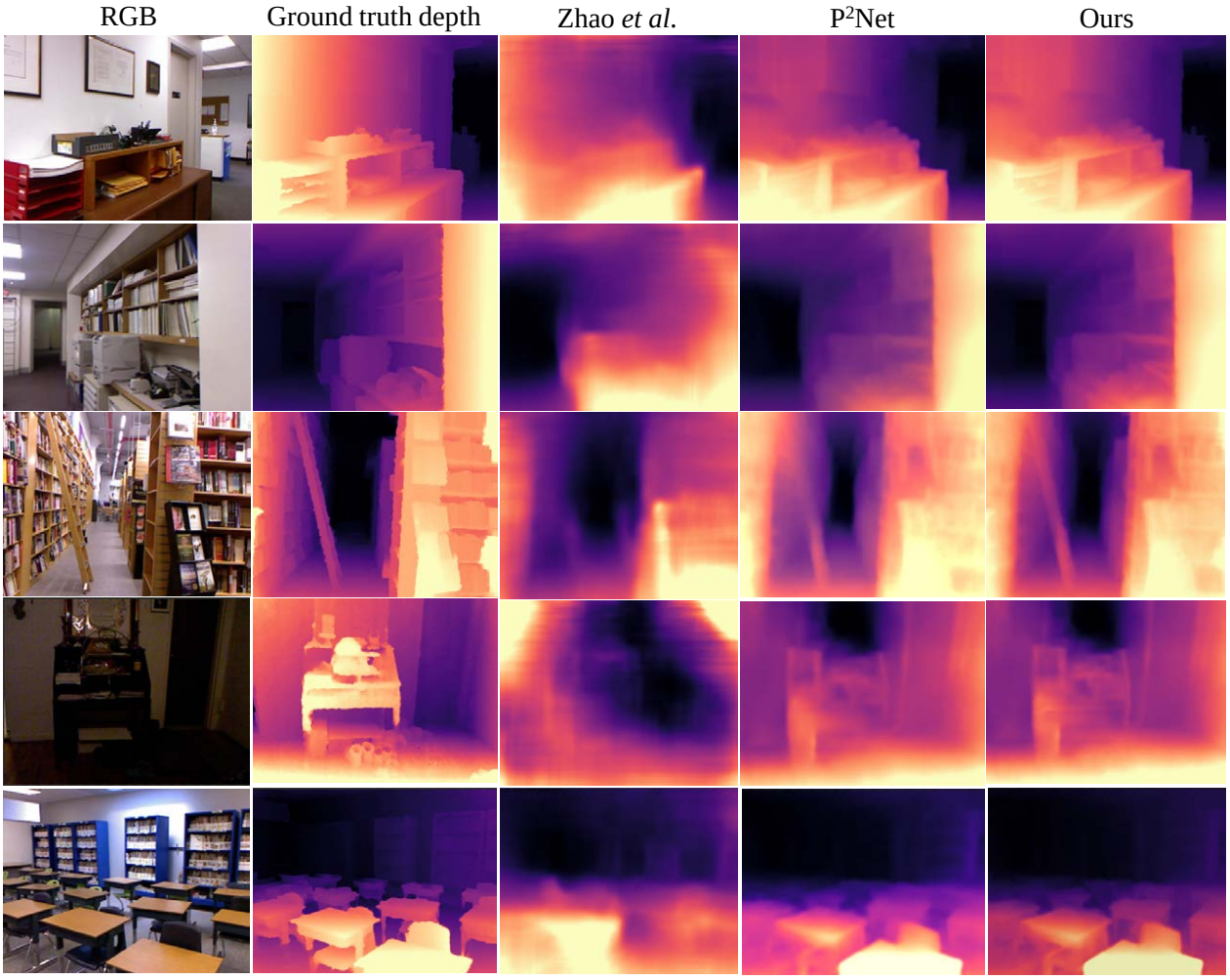
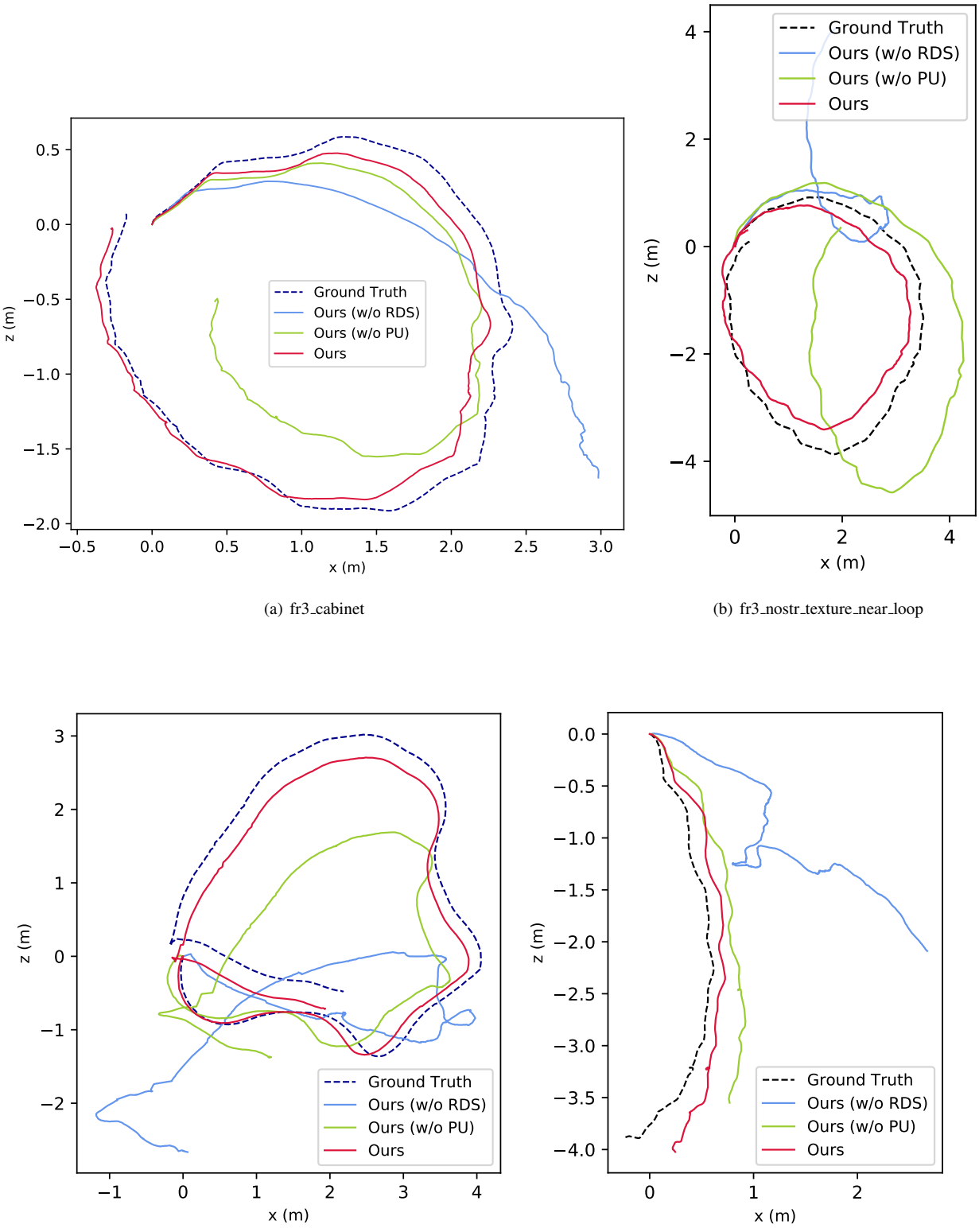


Figure 3. Single-view depth estimation results on NYUv2 dataset.



(a) fr3_cabinet

(b) fr3_nostr_texture_near_loop

(c) fr2_pioneer_360

(d) fr3_str_notexture_far

Figure 4. Ablation studies on various versions of our method that are pretrained on KITTI and tested on TUM dataset. w/o RDS: without refined depth for self-supervision, w/o RU: without predicted photometric uncertainty map.