PLGSLAM: Progressive Neural Scene Represenation with Local to Global Bundle Adjustment (Supplementary Material)

1. Overview

In this supplementary material, we provide the implementation details in Sec.2. In Sec.3 we present the extensive experiments on various datasets. More ablation studies are also presented in Sec. 3.2 to demonstrate the effectiveness of our design choice.

2. Implementation Details

Here we report the detailed settings and hyperparameters used in PLGSLAM to achieve high-quality surface reconstruction, accurate camera tracking and real-time performance.

The truncation distance T is set to 6 cm in our method. The coarse feature planes is employed with a resolution of 24 cm. We use a resolution of 6 cm. All feature planes have 32 channels, resulting in a 64-channel concatenated feature input for the decoders. The decoders are two-layer MLPs with 32 channels in the hidden layer. The dimension of the geometric feature $\mathbf{z}$ is 15. ReLU activation function is used for the hidden layer, and Tanh and Sigmoid are respectively used for the output layers of TSDF and raw colors. We use 16 bins for One-Blob encoding of each dimension. For Replica [6] dataset, we sample N = 32 points for stratified sampling and $N_{surface}$ = 8 points for importance sampling on each ray. We use 200 iterations for first frame mapping. We perform 10 optimization iterations for mapping and randomly select 4000 rays for each iteration. For camera tracking, 2000 rays are chosen at random and 8 optimization iterations are performed. And for ScanNet [2] dataset, we set N = 48 and $N_{surface}$ = 8. Also, we perform 30 optimization iterations for both mapping and tracking in ScanNet scenes. For the scenes in Apartment dataset [10], we similarly set N = 48 and $N_{surface}$ = 8. For this dataset, We perform 30 optimization iterations for mapping and tracking, and we randomly sample 5000 rays for each iteration.

We use different set of loss coefficients for mapping and tracking. We set $\lambda_{fs} = 5$, $\lambda_{sdf_m} = 200$,$\lambda_{sdf_t} = 10$, $\lambda_{smooth} = 0.01$, $\lambda_{re} = 5$, $\lambda_{nw} = 5$, $\lambda_d = 0.1$, and $\lambda_c = 5$. And during tracking, we set $\lambda_{fs} = 10$, $\lambda_{sdf_m} = 200$, $\lambda_{sdf_t} = 50$, $\lambda_{re} = 10$, $\lambda_{nw} = 10$, $\lambda_d = 1$, and $\lambda_c = 5$.

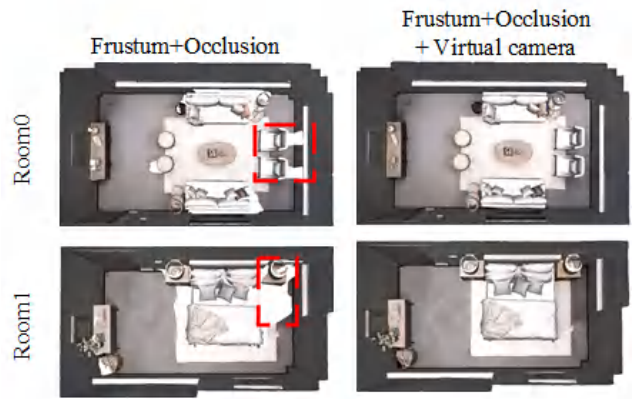

Figure 1. Visualization of different culling strategy applied on gt mesh. Frustum+occlusion culling method removes occluded regions inside the room which results in too many holes in mesh. The frustum+occlusion+ virtual camera method could remove unwanted artefacts outside the room but preserve the completeness of the mesh.

We allocate the mapping process every 4 input frames and use a temporal window of W = 20 keyframes for jointly optimizing the feature tri-planes, MLP decoders, and camera poses of the selected keyframes. We use Adam [4] for optimizing all learnable parameters of our method. We present our progressive scene representation method in Algorithm 1.

Once all input frames are processed, and for evaluation purposes, we build a TSDF volume for each scene and use the marching cubes algorithm [5] to obtain 3D meshes. We use inverse distance weight to fuse our local scene representation into the entire mesh.

2.1. Evaluation

Culling Method. In previous NeRF-based SLAM method, all of them use an extra mesh culling step before evaluating the reconstructed mesh. iMAP [7] and NICE-SLAM [10] adopt a frustum culling strategy which removes the mesh vertices outside any of the camera frustum. This culling strategy remove the artifacts outside

Algorithm 1 Our Progressive Scene Representation.

$j \leftarrow 1$; /* The first scene representation index*/
$p \leftarrow 1$; /* First camera pose index*/
$q \leftarrow 1$; /*Start with the first frames */
$\{R,T\}_{p..q} \leftarrow 1$; /* Initialize poses as identity*/
$\theta_j \leftarrow InitializeSR()$; /* Initialize the first local scene representation*/
while $q < P$ **do**
  while $m = 0; m < 200$ **do**
    Optimize (θ_j); /* Optimize the first local scene representation with the first frame*/
    $m \leftarrow m + 1$;
  end
  while $\{R,t\}_j < Bound_j$ **do**
    $q \leftarrow q + 1$;
    $\{R,T\}_q \leftarrow \{R,T\}_{q-1}$; /* Append a pose at the end of the trajectory */
    Optimize($\{R,T\}_{p..q}, \theta_j$); /* Refine poses and local scene representation */
  end
  if $q < P$ **then**
    $j \leftarrow j + 1$; /* A new scene representation index*/
    $\theta_j \leftarrow InitializeSR()$ /* Initialize a new local scene representation*/
    $t_j \leftarrow t_q$; /* Centered around the last pose*/
    $p \leftarrow p + n$; /* Stop considering the frames in last scene representation*/
  end if
  Repeat until all frames are registered.
end

camera frustum but cannot remove artifacts inside camera frustum. In NeuralRGBD [1] and ESLAM [3], they adopt *frustum+occlusion* culling method. While this strategy could effectively remove some artifacts, their overly aggressive culling strategy results in many holes in the culled mesh. Follow [8], We introduce a modification to the culling strategy used for the quantitative evaluation of the reconstruction accuracy, which leads to a fairer comparison. We use the *frustum+occlusion+virtul camera* culling method. This method simulates virtual camera views that cover the occluded regions. We show the comparison of our method with others in Fig. 5. Our method effectively fills the holes generated by the previous culling strategy.

Evaluation Metrics. After mesh culling, we evaluate the reconstructed mesh with a mixture of 3D (Accuracy, Completion and Completion Ratio) and 2D (Depth L1) metrics. In Tab. 1, we present the 3D reconstruction metrics. We first uniformly sample two point clouds P and Q from both GT and reconstructed meshes, with $|P| = |Q| = 200000$. Accuracy metric is defined as the average distance between a point on GT mesh to its nearest point on reconstructed mesh. The Completion metric is defined as the average distance between a point on reconstructed mesh to its nearest point on GT mesh. The completion metric refers to the proportion of the overall ground truth (GT) where the average distance between a point on the reconstructed mesh and its nearest point on the GT mesh is less than 5 cm.

Reconstruction Metrics	Definition
Depth L1	$\frac{1}{N}\sum(\lvert d_i - d_i^*\rvert)/d_i$
Accuracy	$\sum_{p\in P}(\min_{q\in Q}\lVert p-q\rVert)/\lvert P\rvert$
Completion	$\sum_{q\in Q}(\min_{p\in P}\lVert p-q\rVert)/\lvert Q\rvert$
Completion Ratio	$\sum_{q\in Q}(\min_{p\in P}\lVert p-q\rVert < 0.05)/\lvert Q\rvert$

Table 1. Definitions of scene reconstruction metrics used for evaluation of surface reconstruction quality.

For depth L1 metric, we render depth from $N = 1000$ virtual view of GT and reconstructed mesh. The virtual views are sampled uniformly within the room. Views that have unobserved points will be rejected and re-sampled. Then depth L1 is defined as the average L1 difference between rendered GT depth and rendered depth.

3. Experimental Results

In this section, we conduct various experiments to show the robustness of our method in different experimental settings and to validate our architecture design choices.

3.1. Frame Loss Robustness

We simulate the frame loss on Replica[6] datasets. We randomly remove some frames in these two datasets. We only skip one frames for every interval, such as ID 20 to 22, ID 110 to 112. In Table 3, it can be seen that iMAP [7], NICE-SLAM [10], eslam [3], [8] struggles to recover camera pose

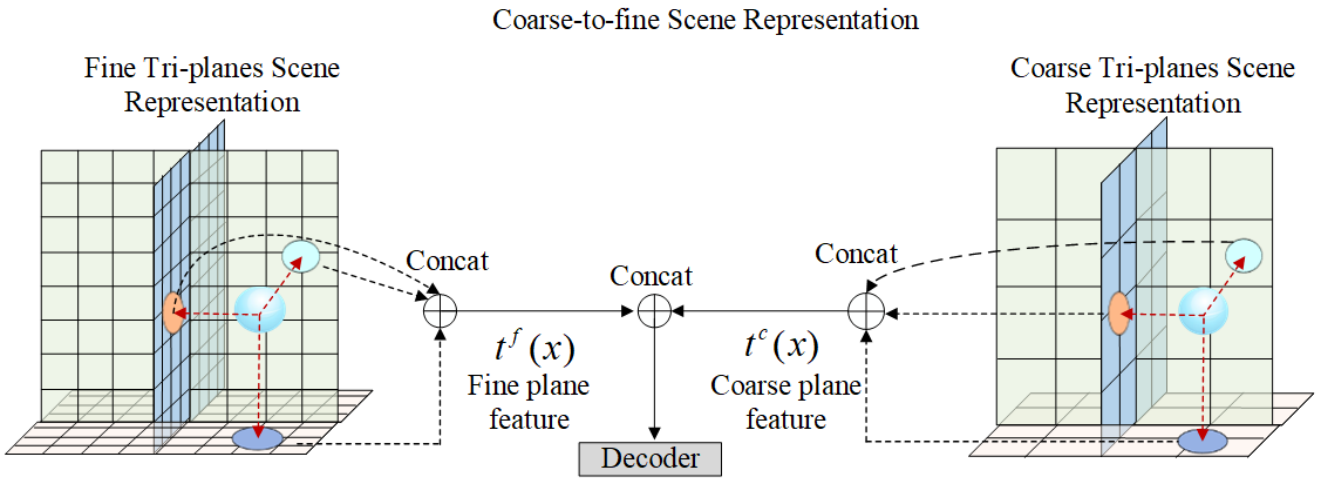

Figure 2. We present our coarse-to-fine local scene representation method. We incorporate a coarse and a fine tri-planes to encode the scene and concate the coarse and fine feature. Then we put them into the decoder to get the sdf and color.

		room0	room1	room2	office0	office1	office2	office3	office4	Avg.
iMap [7]	Depth L1 [cm]	5.08	3.44	5.78	3.79	3.76	3.97	5.61	5.71	4.64
	Acc. [cm]	4.01	3.04	3.84	3.34	3.34	4.06	4.20	4.34	3.62
	Comp.[cm]	5.84	4.40	5.07	3.62	3.62	4.73	5.49	6.65	4.93
	Comp. Ratio(%)[5cm]	78.34	85.85	79.40	83.59	88.45	79.73	73.90	74.77	80.50
NICE-SLAM [10]	Depth L1 [cm]	1.79	1.33	2.20	1.43	1.58	2.70	2.10	2.06	1.90
	Acc. [cm]	2.44	2.10	2.17	1.85	1.56	3.28	3.01	2.54	2.37
	Comp.[cm]	2.60	2.19	2.73	1.84	1.82	3.11	3.16	3.61	2.63
	Comp. Ratio(%)[5cm]	91.81	93.56	91.48	94.93	94.11	88.27	87.68	87.23	91.13
Vox-Fusion[9]	Depth L1 [cm]	1.76	2.52	3.58	3.44	1.77	3.52	1.82	4.84	2.91
	Acc. [cm]	1.77	1.51	2.23	1.63	1.44	2.09	2.33	2.02	1.88
	Comp.[cm]	2.69	2.31	2.58	1.87	1.66	3.03	2.81	3.51	2.56
	Comp. Ratio(%)[5cm]	92.03	92.47	90.13	93.86	94.40	88.94	89.10	86.53	90.93
Co-SLAM[8]	Depth L1 [cm]	1.05	0.85	2.37	1.24	1.48	1.86	1.66	1.54	1.51
	Acc. [cm]	2.11	1.68	1.99	1.57	1.31	2.84	3.06	2.23	2.10
	Comp.[cm]	2.02	1.81	1.96	1.56	1.59	2.43	2.72	2.52	2.08
	Comp. Ratio(%)[5cm]	95.26	95.19	93.58	93.58	94.65	91.63	90.72	90.44	93.44
ESLAM[3]	Depth L1 [cm]	0.73	0.74	1.26	0.71	1.02	0.93	1.03	1.18	0.95
	Acc. [cm]	2.45	2.44	1.70	1.48	1.60	2.55	2.38	2.06	2.08
	Comp.[cm]	1.79	1.58	1.61	**1.30**	1.47	2.05	2.18	2.05	1.75
	Comp. Ratio(%)[5cm]	97.29	96.80	96.89	**98.45**	96.04	96.14	95.33	94.53	96.43
Ours	Depth L1 [cm]	**0.61**	**0.62**	**1.01**	**0.63**	**0.90**	**0.77**	**0.83**	**0.88**	**0.77**
	Acc. [cm]	**2.23**	**2.21**	**1.51**	**1.41**	**1.42**	**2.21**	**2.02**	**1.80**	**1.79**
	Comp.[cm]	**1.49**	**1.38**	**1.54**	**1.30**	**1.27**	**1.85**	**1.91**	**1.85**	**1.54**
	Comp. Ratio (%) [5cm]	**98.39**	**97.51**	**98.52**	98.44	**97.77**	**97.84**	**96.98**	**97.03**	**97.87**

Table 2. Per-scene quantitative results on Replica [6] dataset. Our method achieves consistently better reconstruction in comparison to iMAP [7], NICE-SLAM [10], Vox-FUsion [9], Co-SLAM [8], ESLAM [3].

and scene geometry. Their camera tracking accuracy drop dramatically. In contrast, our method perform better robustness compared with existing methods.

3.2. Ablation Study

In this section, we provide more detailed ablation studies of our method. Our experimental results demonstrate the effectiveness of our method. The joint scene representa-

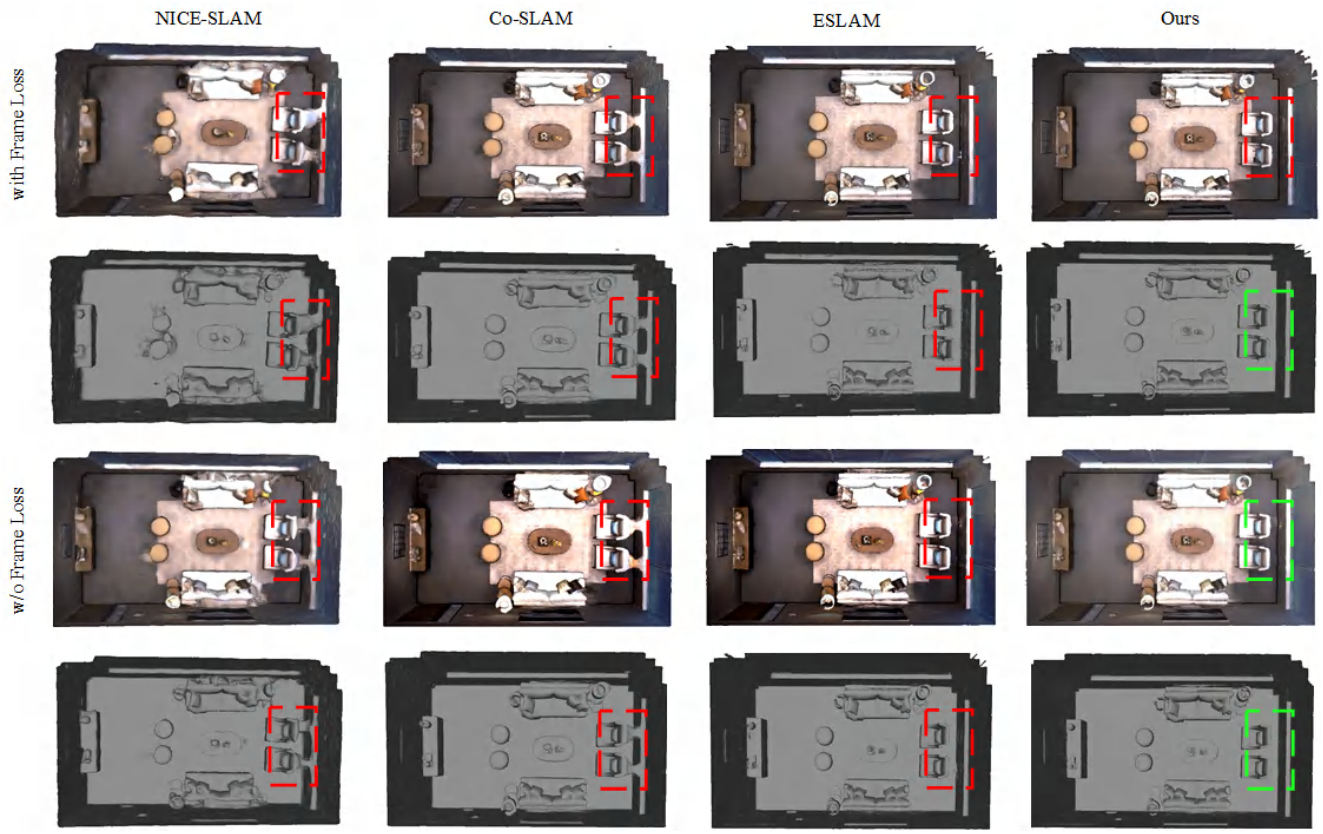

Figure 3. Qualitative comparison of our proposed PLGSLAM method's surface reconstruction and localization accuracy with existing NeRF-based dense visual SLAM methods, NICE-SLAM [10], Co-SLAM [8], and ESLAM [3] on the replica dataset [**?**] with frame loss and without frame loss. The region outlined on the image is marked in red to signify lower predictive accuracy, in green to signify higher accuracy.

tion with tri-planes and MLP significantly improves surface reconstruction accuracy, the adoption of a single network for the entire scene enhances pose estimation and reconstruction metrics, and the inclusion of local-to-global bundle adjustment substantially improves robustness and accuracy in camera tracking. The results demonstrate that our method yields more accurate, smooth, and true-to-reality reconstructions.The detailed experimental results is shown in Tab. 4.

3.3. Extensive Experiments

The extensive experiments show that our PLGSLAM method performs well on different datasets. Specifically, on the Replica dataset, our view synthesis results surpass those of NICE-SLAM, showing that our approach creates more accurate and realistic visualizations. Additionally, on the apartment dataset, our PLGSLAM method outperforms other existing NeRF-based SLAM system such as Co-SLAM and ESLAM, proving its superior surface reconstruction ability.

Methods	Metrics	Room0*	Room0
iMAP* [7]	RMSE[m]	0.315	0.0552
	Mean[m]	0.247	0.0353
NICE-SLAM [10]	RMSE[m]	0.183	0.0223
	Mean[m]	0.126	0.0174
Co-SLAM [8]	RMSE[m]	0.033	0.0072
	Mean[m]	0.023	0.0083
ESLAM [3]	RMSE[m]	0.045	0.0065
	Mean[m]	0.018	0.0072
Ours	RMSE[m]	0.030	0.0060
	Mean[m]	0.020	0.0067

Table 3. Camera tracking results on Replica [6] dataset with frame loss. Our method achieves better camera tracking results in comparison to iMAP [7], NICE-SLAM [10], Vox-Fusion [9], Co-SLAM [8], ESLAM [3] in frame loss experiments.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface

Methods	Reconstruction[cm]			Localization[cm]	
	Accuracy	Completion	Completion Ratio(%)	Mean	RMSE
with only tri-planes scene representation	13.314	4.68	82.34	5.935	6.787
with only coarse planes and remove the fine planes	15.765	5.68	78.47	5.982	6.879
with only fine planes and remove the coarse planes	15.473	5.57	79.95	5.882	6.741
replacing the concatenating with summation	11.873	4.25	84.37	5.683	6.527
with only MLP scene representation	18.754	10.74	63.54	8.973	9.317
w/o progressive scene representation	12.754	4.23	83.15	5.875	6.693
w/o overlapping	12.772	4.03	85.27	5.793	6.605
without depth guided sampling	12.475	4.08	85.67	5.724	6.531
with only one loss function for the whole truncation region	12.087	4.23	81.35	5.681	6.504
no camera pose optimization during mapping	13.192	4.88	80.34	5.925	6.742
with only local BA	12.435	4.18	83.47	5.774	6.391
with only global BA	13.741	4.68	82.19	5.884	6.597
only use BA loss in tracking thread	13.471	4.57	82.54	5.947	6.579
only use BA loss in mapping thread	12.762	4.19	83.75	5.932	6.453
Ours	**11.086**	**3.26**	86.55	5.574	6.228

Table 4. Detailed ablation study. We conduct experiments on Apartmentdataset [10] to verify the effectiveness of our method. Our fullmodel achieves better completion reconstructions and more accurate pose estimation results.

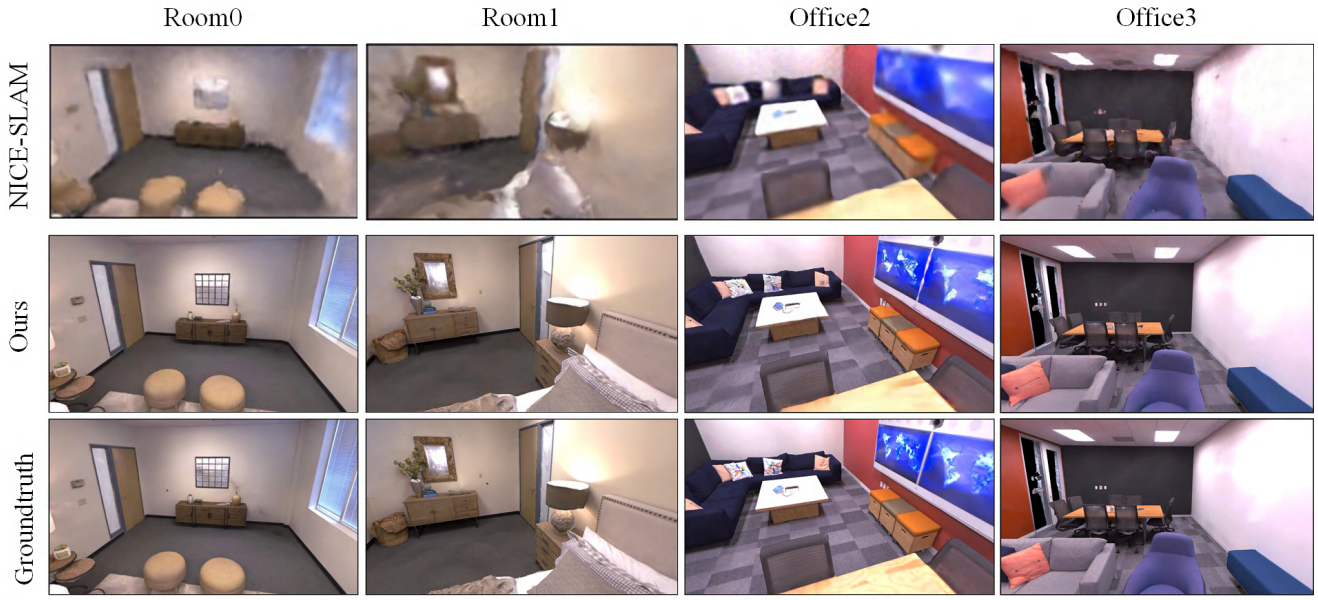

Figure 4. Qualitative comparison of our proposed PLGSLAM method's view synthesis results with existing NeRF-based dense visual SLAM methods, NICE-SLAM [10] on the Replica dataset [6].

reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6290–6301, June 2022. 2

[2] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[3] Mohammad Mahdi Johari, Camilla Carta, and François Fleuret. Eslam: Efficient dense slam system based on hybrid representation of signed distance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17408–17419, 2023. 2, 3, 4, 6

[4] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 1

[5] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. 1

[6] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik

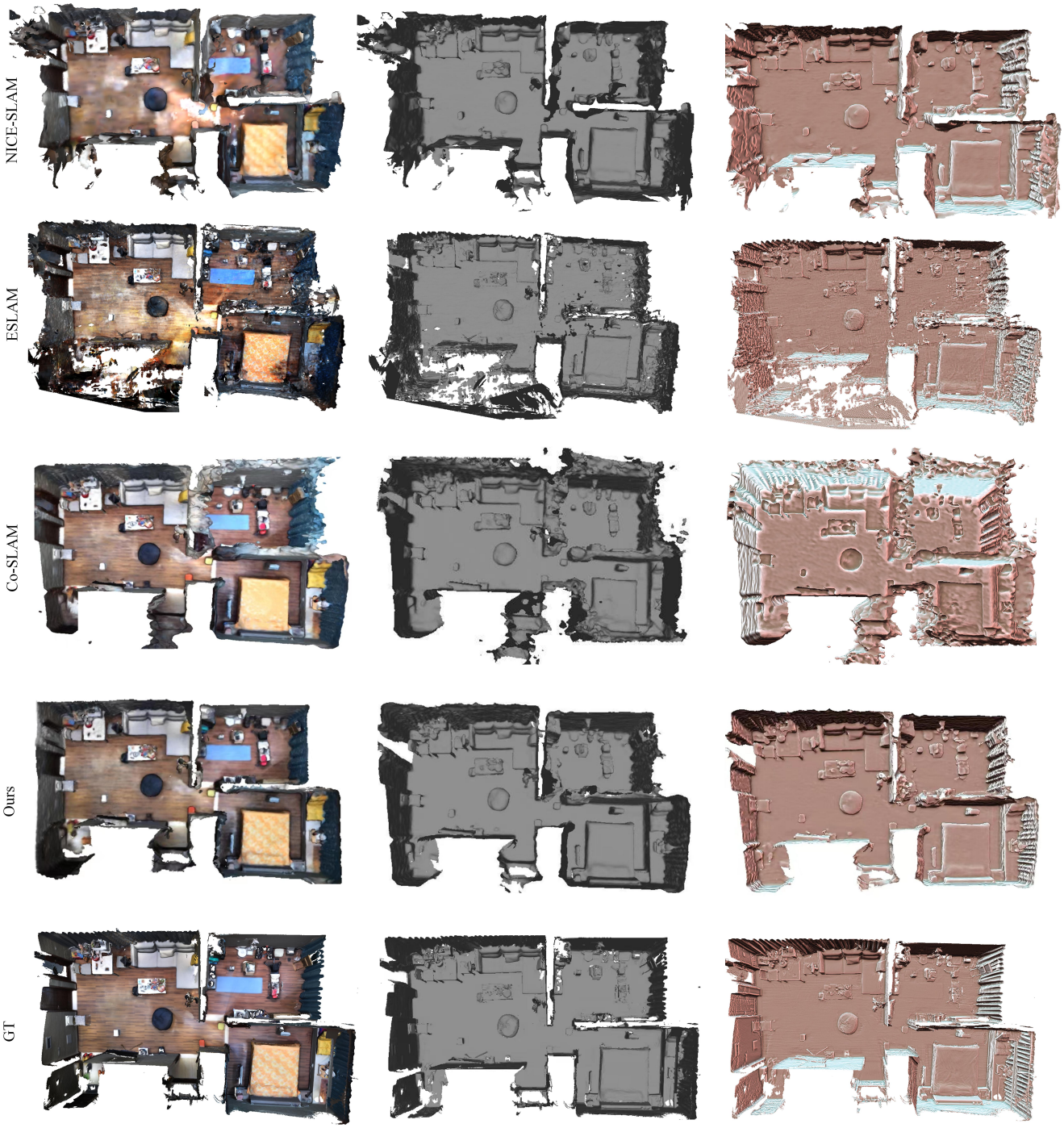

Figure 5. Qualitative comparison of our proposed PLGSLAM method's surface reconstruction and localization accuracy with existing NeRF-based dense visual SLAM methods, NICE-SLAM [10], Co-SLAM [8], and ESLAM [3] on the apartment dataset [10].

Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 1, 2, 3, 4, 5

[7] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J. Davison. imap: Implicit mapping and positioning in real-time. In *ICCV*, pages 6229–6238, October 2021. 1, 2, 3, 4

[8] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13293–13302, 2023. 2, 3, 4, 6

[9] Xingrui Yang, Hai Li, Hongjia Zhai, Yuhang Ming, Yuqian Liu, and Guofeng Zhang. Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 499–507, 2022. 3, 4

[10] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *CVPR*, pages 12786–12796, June 2022. 1, 2, 3, 4, 5, 6