Supplementary Material for
NICE-SLAM: Neural Implicit Scalable Encoding for SLAM

Zihan Zhu[1,2*] Songyou Peng[2,4*] Viktor Larsson[3] Weiwei Xu[1] Hujun Bao[1]
Zhaopeng Cui[1†] Martin R. Oswald[2,5] Marc Pollefeys[2,6]

[1]State Key Lab of CAD&CG, Zhejiang University [2]ETH Zurich [3]Lund University
[4]MPI for Intelligent Systems, Tübingen [5]University of Amsterdam [6]Microsoft

In the supplementary material we present the following:

- Implementation details and parameters (Section A)
- Additional experiments and ablations (Section B)

A. Implementation Details

A.1. Frustum Feature Selection

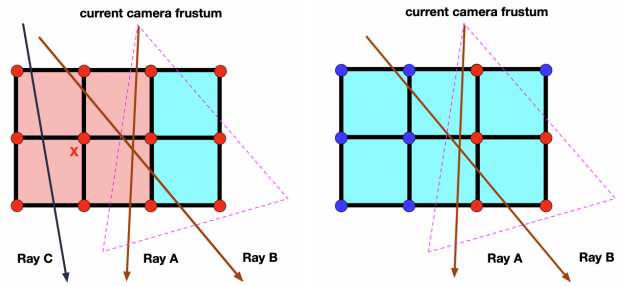
The grid-based representation allows us to only optimize the geometry within the current viewing frustum while keeping the rest of the scene geometry fixed. However, naive optimization for all voxels will affect features even just slightly outside the viewing frustum because of trilinear interpolation. This is illustrated in Fig. Aa. The rays *A* and *B* are viewing rays from the current frame and an active keyframe, respectively. Including these rays in the optimization will update the feature at *X* (marked in the figure) due to trilinear interpolation. However, updating this feature will also affect the ray *C* coming from an inactive keyframe.

To solve the problem, we propose to only update features fully inside the current viewing frustum during the optimization, see Fig. Ab. In this way, it will not only preserve the previously reconstructed geometry, but also significantly reduce the number of parameters during optimization.

A.2. Hierarchical Feature Grid Initialization

Coarse-level Feature Grid. The coarse-level feature grid is randomly initialized in all experiments.

Mid-level Feature Grid. The mid-level feature grid is also randomly initialized in all experiments, except for the result shown in Fig. 7 in the main paper, where it is initialized to free space to better visualize the predictions from the coarse-level grid. Empirically we find that the random initialization gives slightly better convergence compared to initializing from a fixed feature vector corresponding to the free space.



(a) Interpolation problem. (b) Feature selection.

Figure A. 2D illustration of the feature grid. The lattice points correspond to features. The optimized and fixed features are shown in red and blue respectively.

Fine-level Feature Grid. The fine-level feature grid is initialized to ensure the output of the fine-level decoder f^2 as zero, as it is added in a residual manner onto the occupancy predicted from the mid-level features. This guarantees a smooth energy transition in the coarse-to-fine optimization. During the training of the fine-level decoder from ConvONet [4], we add additional regularization loss to enforce that, if the fine-level feature is zero, no matter what the concatenated mid-level feature is, the output residual should always be zero. This regularization allows us to zero-initialize the fine-level grid at runtime.

A.3. Justification for Design Choices.

Why 3-level Feature Grids? We show in Fig. 8 in the main paper that using hierarchical grids leads to better convergence compared to a single level, and we find that the current design guarantees a good balance between the quality and real-time capability / memory consumption (only 12 MB for Replica scenes). We also conduct an ablation study

*Equal contribution.
†Corresponding author.

Levels	2	3	4
FLOPs [$\times 10^3$] $\downarrow$	58.45	104.16	155.95
Depth L1 [cm] $\downarrow$	**1.86**	1.87	1.96
Acc. [cm] $\downarrow$	2.87	**2.78**	3.15
Comp. [cm] $\downarrow$	2.76	2.76	**2.40**
Comp. Ratio [$<$ 5cm %] $\uparrow$	91.24	91.37	**93.60**

Table A. **Ablation on the Levels of Feature Grids.** Reconstruction results on Replica `room-0` with ground truth camera pose.

on the number of levels of feature grids in Table A. It shows that the 3-level feature grid is a good balance between the reconstruction quality and computational efficiency.

Why is the Mid-level Output not a Residual to the Coarse-level Output? The coarse grid has a significantly larger voxel size (side of > 1 meter) than the mid and fine levels, so updating the coarse-level feature would affect a large area. To ensure small local updates for efficiency, we disconnect coarse level from mid and fine levels, and only use coarse level for prediction.

A.4. Mesh Visualization

The reconstructed scene is represented implicitly using hierarchical feature grids. We use the marching cubes algorithm [3] to create a mesh for the visualization purpose. For every observed point we predict its occupancy value using the fine-level decoder and color from the color decoder. For those unseen points in the predicted regions (i.e. voxels with partial observations in the coarse grid), we predict occupancy from the coarse-decoder and set the color to cyan for visualization as shown in Fig.8 in our main paper and the supplementary video. Other points are assigned zero occupancy. The same resolution is used in marching cubes for both iMAP* and NICE-SLAM.

A.5. Decoder Pretraining

We use the Synthetic Indoor Scene Dataset provided in ConvONet [4] to pre-train the encoder-decoder. Furthermore, we use the Point Cloud Encoder instead of the Voxel Encoder. All levels are trained with room_grid64 setting in ConvONet [4]. The feature dimension for all the feature grids is 32. As for hyperparameters used for the pretraining process, we follow the same setting as ConvONet [4].

A.6. Hyperparameters

Here we report detailed hyperparameters of online tracking and mapping used for both NICE-SLAM and iMAP*. We perform tracking for every frame and optimize the geometry every fifth frame, except for TUM RGB-D where we optimize the geometry every frame. All parameters are tuned to keep a good balance between the accuracy and the efficiency.

Mapping Iterations	15	30	60	120	240
Depth L1 [cm] $\downarrow$	2.31	2.03	1.87	1.74	1.59
Acc. [cm] $\downarrow$	2.90	2.84	2.78	2.80	2.78
Comp. [cm] $\downarrow$	3.14	2.91	2.76	2.65	2.50
Comp. Ratio [$<$ 5cm %] $\uparrow$	89.15	90.55	91.37	91.94	92.76

Table B. **Ablation on Mapping Iterations.** Reconstruction results on Replica `room-0` with ground truth camera poses.

NICE-SLAM. For scene geometry optimization, we use a maximum of 60 iterations for all datasets. In terms of tracking, we use 10 iterations for small-scale synthetic datasets (Replica and Co-Fusion). For the large-scale real datasets including ScanNet and our self-captured scene, we use 50 iterations for tracking. For TUM RGB-D dataset we use 200 iterations.

The learning rate for tracking on Replica [6], TUM RGB-D [7], ScanNet [1], Self-captured, and Co-Fusion [5] are $1e{-}3$, $1e{-}2$, $5e{-}4$, $3e{-}3$, $1e{-}3$ respectively. The learning rate for optimizing the coarse-level is $1e{-}3$, for mid-level is $1e{-}1$, for fine- and color-level is $5e{-}3$. The learning rate for selected keyframes' camera parameters during the mapping is $1e{-}3$, except for Co-Fusion where we set the learning rate to 0.

iMAP*. For all datasets except TUM RGB-D [7], we use 50 iterations for tracking and 300 iterations for joint optimization. For TUM RGB-D [7], we use 200 and 300 iterations respectively. The learning rate for tracking on Replica [6], TUM RGB-D [7], ScanNet [1], Self-captured, and Co-Fusion [5] are $5e{-}4$, $5e{-}3$, $2e{-}3$, $1e{-}3$, $5e{-}4$ respectively. The learning rate for joint optimization is $2e{-}4$.

B. Additional Experiments

B.1. Frame Loss Robustness

We simulate extreme frame loss on ScanNet `scene0000_00` by skipping 100 frames from frame ID 2001 to 2100. As visualized in Fig. B, iMAP* struggles to recover camera poses and scene geometry, even given 1500 iterations. In contrast, our NICE-SLAM is able to recover the camera pose using only 300 iterations. This is due to the use of coarse-level geometric representation which improves the prediction capability.

B.2. Number of Mapping and Tracking Iterations

We show in Fig. C how the number of tracking and mapping iterations affects the tracking performance. We also give ground truth camera pose and evaluate reconstruction with different mapping iterations in Table B.

B.3. Frustum Feature Selection

To highlight the importance of current frustum feature selection (see Section A.1), we run our system with and

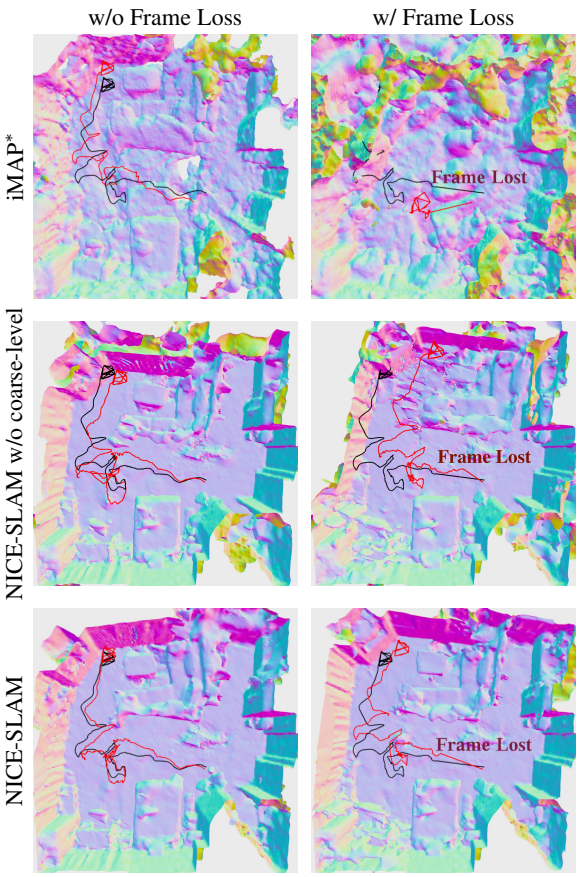


Figure B. **Robustness to Frame Loss.** We show the results at frame 2100 after frame loss at frame 2000. The black trajectory is the ground truth from ScanNet [1], and the red trajectory indicates tracking results. The missing frames corresponds to the straight line in the middle.

without the selection process. The results are shown in Fig. D. Without fixing the border features, significant artifacts appear in the reconstruction (Fig. Aa).

B.4. More Results on Replica Dataset [6]

Here we provide the detailed results for all Replica scenes. Table C shows the quantitative comparison when considering the average metric values for 5 consecutive runs, and only evaluate without unseen regions that are outside all camera's viewing frustums. What is more, as done in [8] we also report the best metrics in 5 consecutive runs under all regions in Table D. As can be noticed, our iMAP re-implementation iMAP* has similar performance over the original iMAP.

In addition, to better highlight the performance differences, we provide additional visualizations using different rendering settings in Fig. E.

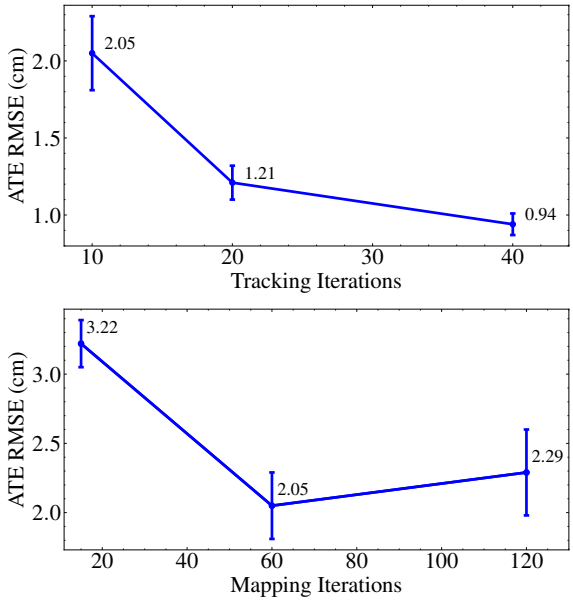


Figure C. **Ablation on the tracking performance.** ATE RMSE (cm) is used as the metric.

B.5. More Results on ScanNet [1]

We show the 3D reconstruction process of iMAP* and NICE-SLAM on ScanNet `scene0000` in Fig. F.

References

[1] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2, 3, 6

[2] Jiahui Huang, Shi-Sheng Huang, Haoxuan Song, and Shi-Min Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8932–8941, 2021. 5

[3] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH*, 1987. 2

[4] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *ECCV*, 2020. 1, 2

[5] Martin Rünz and Lourdes Agapito. Co-fusion: Real-time segmentation, tracking and fusion of multiple objects. In *ICRA*, 2017. 2

[6] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica

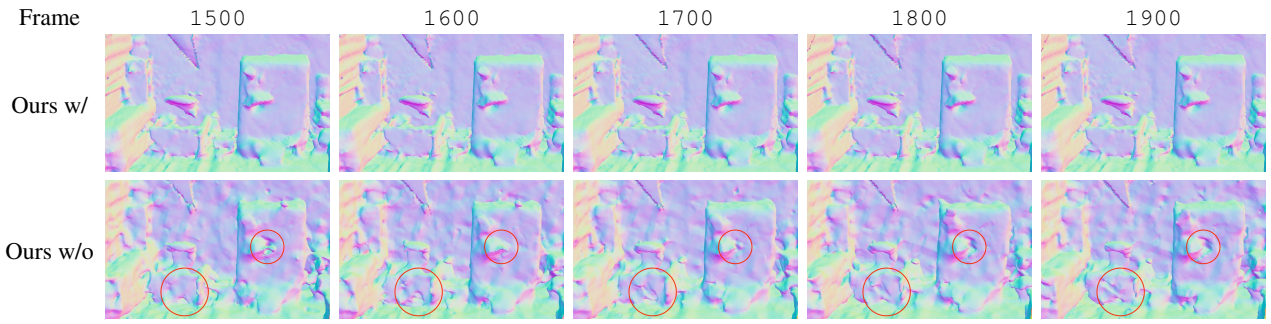


Figure D. **Ablation on Frustum Feature Selection.** We show our method with and without the frustum feature selection run on sequence scene0000_00 in the ScanNet datasets. During these frames the camera is scanning other parts of the scene. The cutout shown in the figure is part of the previously reconstructed geometry and should remain constant. The mesh is visualized with the vertex normal.

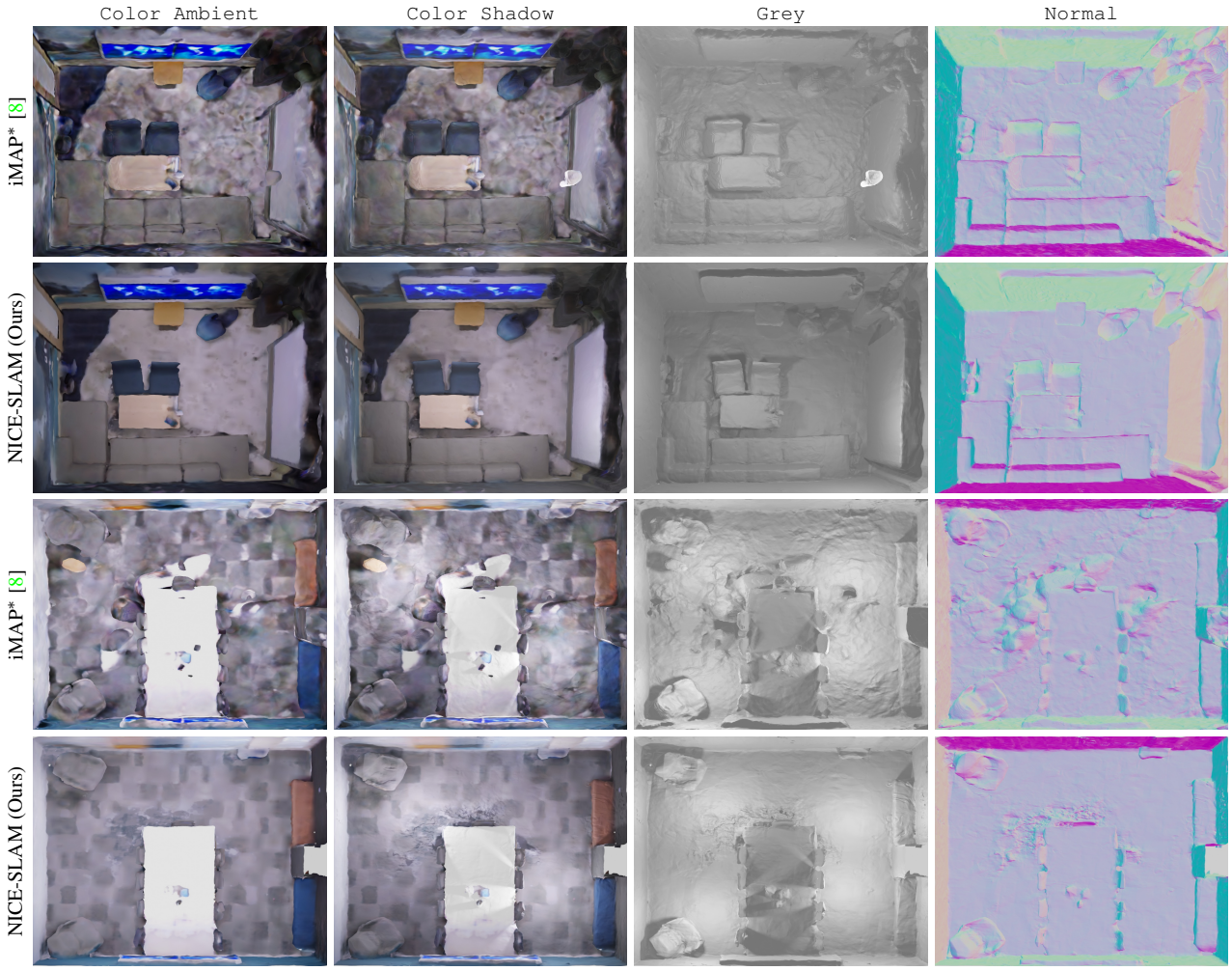


Figure E. **More Results on Replica Dataset [6]**. We visualize the final reconstruction on two scenes including office-0 (top two rows) and office-4 (bottom two rows). To better show the differences, we use different rendering settings. As can be visualized, our NICE-SLAM produces high-quality geometry and colors.

		room-0	room-1	room-2	office-0	office-1	office-2	office-3	office-4	Avg.
TSDF-Fusion Res. = 512 (536.87MB)	**Depth L1** [cm] ↓	6.38	5.33	6.84	4.74	4.62	11.32	9.89	6.49	6.95
	Acc. [cm] ↓	1.87	2.48	1.69	1.14	0.96	1.63	2.08	1.74	1.70
	Comp. [cm] ↓	3.60	3.20	2.85	1.72	2.31	3.66	3.69	3.91	3.12
	Comp. Ratio [< 5cm %] ↑	88.33	89.82	90.38	93.55	90.35	86.74	85.35	86.31	88.85
TSDF-Fusion Res. = 256 (67.10MB)	**Depth L1** [cm] ↓	6.69	5.47	7.47	4.97	5.28	12.30	11.17	7.20	7.57
	Acc. [cm] ↓	1.76	2.11	1.59	1.15	0.97	1.56	1.98	1.66	**1.60**
	Comp. [cm] ↓	3.85	3.36	3.33	1.93	2.68	4.17	4.22	4.37	3.49
	Comp. Ratio [< 5cm %] ↑	86.29	88.44	86.63	91.73	87.88	82.95	81.31	83.38	86.08
iMAP* [8] (1.04MB)	**Depth L1** [cm] ↓	5.70	4.93	6.94	6.43	7.41	14.23	8.68	6.80	7.64
	Acc. [cm] ↓	5.66	5.31	5.64	7.39	11.89	8.12	5.62	5.98	6.95
	Comp. [cm] ↓	5.20	5.16	5.04	4.35	5.00	6.33	5.47	6.10	5.33
	Comp. Ratio [< 5cm %] ↑	67.67	66.41	69.27	71.97	71.58	58.31	65.95	61.64	66.60
DI-Fusion [2] (3.78MB)	**Depth L1** [cm] ↓	6.66	96.82	36.09	7.36	5.05	13.73	11.41	9.55	23.33
	Acc. [cm] ↓	1.79	49.00	26.17	70.56	1.42	2.11	2.11	2.02	19.40
	Comp. [cm] ↓	3.57	39.40	17.35	3.58	2.20	4.83	4.71	5.84	10.19
	Comp. Ratio [< 5cm %] ↑	87.77	32.01	45.61	87.17	91.85	80.13	78.94	80.21	72.96
NICE-SLAM (12.02MB)	**Depth L1** [cm] ↓	2.11	1.68	2.90	1.83	2.46	8.92	5.93	2.38	**3.53**
	Acc. [cm] ↓	2.73	2.58	2.65	2.26	2.50	3.82	3.50	2.77	2.85
	Comp. [cm] ↓	2.87	2.47	3.00	2.02	2.36	3.57	3.83	3.84	**3.00**
	Comp. Ratio [< 5cm %] ↑	90.93	92.80	89.07	94.93	92.61	85.20	82.98	86.14	**89.33**

Table C. **Reconstruction Results for the Replica Dataset (Average of 5 runs).**

		room-0	room-1	room-2	office-0	office-1	office-2	office-3	office-4	Avg.
TSDF-Fusion Res. = 512 (536.87MB)	**Acc.** [cm]	5.20	2.83	1.60	1.66	1.06	2.29	2.50	2.18	2.42
	Comp. [cm]	5.05	4.60	4.50	1.06	9.57	5.84	4.16	4.30	4.89
	Comp. Ratio [< 5cm %]	75.07	79.03	86.01	80.19	77.80	80.69	82.29	83.00	80.51
TSDF-Fusion Res. = 256 (67.10MB)	**Acc.** [cm]	4.17	2.69	1.49	1.65	1.09	2.24	2.37	2.16	**2.23**
	Comp. [cm]	5.65	4.85	5.04	10.88	9.85	6.94	4.93	4.95	6.64
	Comp. Ratio [< 5cm %]	72.99	77.19	83.10	78.52	76.43	75.66	76.74	79.01	77.46
iMAP [8] (1.04MB)	**Acc.** [cm] ↓	3.58	3.69	4.68	5.87	3.71	4.81	4.27	4.83	4.43
	Comp. [cm] ↓	5.06	4.87	5.51	6.11	5.26	5.65	5.45	6.59	5.56
	Comp. Ratio [< 5cm %] ↑	83.91	83.45	75.53	77.71	79.64	77.22	77.34	77.63	79.06
iMAP* [8] (1.04MB)	**Acc.** [cm] ↓	4.07	3.86	5.17	5.40	4.04	5.23	4.30	4.98	4.63
	Comp. [cm] ↓	4.73	4.32	5.53	4.95	5.27	5.40	4.94	5.08	5.03
	Comp. Ratio [< 5cm %] ↑	79.12	76.21	69.19	77.47	76.70	70.53	73.51	71.81	74.32
DI-Fusion [2] (3.78MB)	**Acc.** [cm]	2.02	277.51	24.94	61.73	1.75	2.63	2.97	2.11	46.96
	Comp. [cm]	3.90	82.87	20.16	12.08	8.76	6.89	5.70	5.96	18.29
	Comp. Ratio [< 5cm %]	86.58	24.77	41.50	74.20	79.22	73.36	70.24	78.26	66.02
NICE-SLAM (12.02MB)	**Acc.** [cm]	2.97	3.23	3.46	5.47	3.33	4.40	3.55	2.87	3.66
	Comp. [cm] ↓	3.30	3.07	3.75	4.54	3.83	3.90	4.49	3.91	**3.85**
	Comp. Ratio [< 5cm %] ↑	89.51	86.01	81.14	85.27	88.01	82.61	79.49	85.33	**84.67**

Table D. **Reconstruction Results for the Replica Dataset (Best in 5 runs).** The numbers for iMAP are directly taken from [8].

dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 2, 3, 4

[7] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *IROS*, 2012. 2

[8] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew Davison. iMAP: Implicit mapping and positioning in real-time. In *ICCV*, 2021. 3, 4, 5, 6

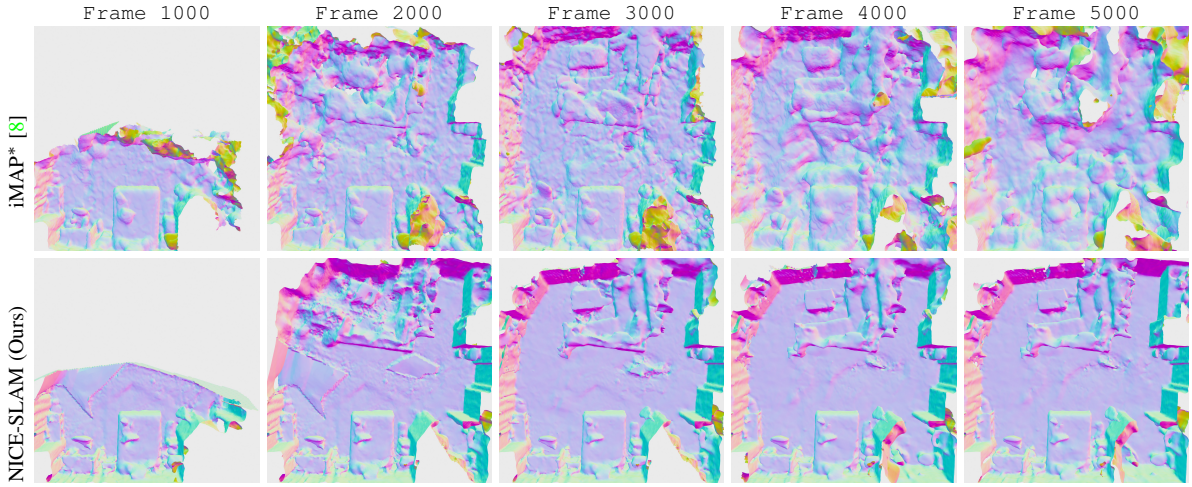


Figure F. **3D Reconstruction Process on ScanNet [1].** Due to our local map updates the resulting geometry is temporally more stable and often less noisy compared to iMAP* [8].