Supplementary Material for: Geometry-guided Feature Learning and Fusion for Indoor Scene Reconstruction

Ruihong Yin[1], Sezer Karaoglu[1,2], Theo Gevers[1,2]
[1]University of Amsterdam, Amsterdam, The Netherlands
[2]3DUniversum, Amsterdam, The Netherlands
r.yin@uva.nl, s.karaoglu@3duniversum.com, Th.Gevers@uva.nl

This supplementary document provides additional details and experimental results of our geometry integration mechanism. Section A presents the definitions of 3D evaluation metrics. Section B details the computation of geometric priors. Details of network implementation are given in Section C. Section D shows additional geometry measures, ablation study, analysis, and visualizations.

A. Evaluation metrics

Table S1 presents the definitions of 3D evaluation metrics in Atlas [4], *i.e*. accuracy (acc), completeness (comp), precision (prec), recall, and F-score. For accuracy and completeness, lower is better. Conversely, for the remaining metrics, higher values correspond to better performance.

3D Metrics	
Acc	$\text{mean}_{p\in P}(\min_{p^*\in P^*}\|\|p-p^*\|\|)$
Comp	$\text{mean}_{p^*\in P^*}(\min_{p\in P}\|\|p-p^*\|\|)$
Prec	$\text{mean}_{p\in P}(\min_{p^*\in P^*}\|\|p-p^*\|\|<.05)$
Recall	$\text{mean}_{p^*\in P^*}(\min_{p\in P}\|\|p-p^*\|\|<.05)$
F-score	$\frac{2\times\text{Prec}\times\text{Recall}}{\text{Prec}+\text{Recall}}$

Table S1. Definitions of 3D metrics. p and p^* are the predicted and ground truth point clouds.

B. Geometric priors

Geometric priors used in our method are introduced in this section.

Viewing direction $\mathbf{v}_i$: The normalized unit direction from the camera origin to the 3D voxel. In our method, it is encoded similarly to NeRF [3], *i.e*.,

$$\begin{aligned}\gamma(\mathbf{v}_i) =&(\sin(2^0\pi\mathbf{v}_i),\cos(2^0\pi\mathbf{v}_i),\dots,\\ &\sin(2^{L-1}\pi\mathbf{v}_i),\cos(2^{L-1}\pi\mathbf{v}_i))\end{aligned} \tag{S-1}$$

where $\gamma(\cdot)$ is applied to each dimension of $\mathbf{v}_i$, and $L=4$ in our experiments.

Projected normal $\mathbf{n}_i$: The vector mapped from the 2D normal according to perspective projection.

Viewing angle $\boldsymbol{\theta}_i$: The absolute value of the cosine similarity between the projected normal and the viewing direction.

Projected depth $\mathbf{z}_i$: The perpendicular distance from the 3D voxel to the camera center. Our method divides the distance by the maximum depth $D_{max}=3$m. The projected depth is also encoded using Eq. S-1 with $L=4$.

Relative pose distance: The pose distance between two cameras. The overall pose distance between camera j and camera k is calculated by

$$\begin{aligned}rp_{jk} &= \sqrt{||\mathbf{t}_{jk}||^2+\frac{2}{3}\text{tr}(\mathbb{I}-\mathbf{R}_{jk})}\\ &= \sqrt{rp(\mathbf{t}_{jk})^2+rp(\mathbf{R}_{jk})^2}\end{aligned} \tag{S-2}$$

where $\mathbf{t}_{jk}$ is the relative translation matrix, $\mathbf{R}_{jk}$ is the relative rotation matrix. $rp(\mathbf{t}_{jk})$ denotes the pose translation distance. $rp(\mathbf{R}_{jk})$ denotes the pose rotation distance. *tr* is the matrix trace operator. In our proposed geometry-guided adaptive feature fusion, $rp(\mathbf{R}_{jk})$, $rp(\mathbf{t}_{jk})$, and rp_{jk} are all used as priors to learn the weight function.

Projective occupancy: In a camera coordinate frame, the projective TSDF $S_p(\mathbf{p})$ of a voxel is the truncated signed distance between the voxel $\mathbf{p}$ and the nearest surface. Projective occupancy $O(\mathbf{p})$ and visibility $V(\mathbf{p})$ are functions of projective TSDF, which can be written by

$$\begin{aligned}O(\mathbf{p}) &= [|S_p(\mathbf{p})|<t] \equiv \begin{cases}1, & |S_p(\mathbf{p})|<t\\ 0, & |S_p(\mathbf{p})|\geq t\end{cases}\\ V(\mathbf{p}) &= [S_p(\mathbf{p})\geq 0] \equiv \begin{cases}1, & S_p(\mathbf{p})\geq 0\\ 0, & S_p(\mathbf{p})<0\end{cases}\end{aligned} \tag{S-3}$$

where t is the truncation distance.

	Prec ↑	Recall ↑	F-score↑
NeuralRecon + G2FL (w/o)	0.697	0.530	0.600
NeuralRecon + G2FL(w)	0.701	0.530	0.602

Table S2. Ablation study for the encoding function defined in Eq. S-1. *w/o* and *w* mean without and with the encoding function respectively.

	$P_{11.25}$ ↑	$R_{11.25}$ ↑	$P_{22.5}$ ↑	$R_{22.5}$ ↑	P_{30} ↑	R_{30} ↑
NeuralRecon	0.501	0.418	0.697	0.608	0.764	0.679
NeuralRecon + ours	**0.581**	0.504	**0.753**	0.680	**0.809**	0.742
VoRTX	0.515	0.479	0.698	0.663	0.757	0.726
VoRTX + ours	0.552	**0.528**	0.719	**0.701**	0.777	**0.761**

Table S3. 3D normal evaluation.

C. Implementation details

In our geometry-guided feature learning, the MLP is composed of 2 linear layers and 2 ReLUs, with channel sizes [37, 32, 1]. The 37 input channels consist of 3×8 encoded viewing directions, 1×8 encoded projected depths, 3 viewing directions, 1 projected depth, and 1 viewing angle. The channel size of linear layer in $\mathcal{T}_1$ is $[C_v + 1, C_v]$. The linear layer in $\mathcal{T}_2$ has $[C_v + 3, C_v]$ channels.

In our geometry-guided adaptive feature fusion, the MLP includes 3 linear layers, with two ReLUs following the first two linear layers. The channel sizes are [81, 32, 32, 9], where the input channel 81 is composed of 2×9 for the mean and standard deviation of the attention matrix, 6×9 for the mean and standard deviation of the relative pose distance, and 1×9 for the occlusion prior. The channel of the linear layer for projective occupancy prediction is $[C_v, 1]$.

In the ablation study for our consistent 3D normal loss, the Gaussian function in Table 5e is defined by $\exp(-\frac{(s_{2d3d}(\mathbf{p})-1)^2}{\sigma^2})$, in which $\sigma^2 = 0.5$.

D. Additional results

3D normal estimation. To further demonstrate that our approach can reconstruct more accurate geometry, this work also provides fine-grained geometry measures, *i.e*. precision P_τ and recall R_τ of the 3D normal, see Eq. S-4 and Eq. S-5. The 3D vertex normals are generated from meshes, and evaluated following the metrics in [1]. The experimental results in Table S3 demonstrate that our proposed geometry integration mechanism enhances the normal performance, thereby contributing to the reconstruction of more precise and accurate geometry.

$$\begin{aligned} P_\tau &= \text{mean}_{p \in P}(\text{angle}(p, p^*) < \tau), \\ p^* &= \min_{p^* \in P^*} ||p - p^*|| \end{aligned} \tag{S-4}$$

$$\begin{aligned} R_\tau &= \text{mean}_{p^* \in P^*}(\text{angle}(p, p^*) < \tau), \\ p &= \min_{p \in P} ||p - p^*|| \end{aligned} \tag{S-5}$$

where p and p^* are the predicted and ground truth points. $angle$ is computed between ground truth and prediction. τ is angle threshold, $\tau \in \{11.25°, 22.5°, 30°\}$.

Additional ablation study. Table S2 validates the effectiveness of the NeRF-like encoding function in our geometry-guided feature learning. As can be seen, with the encoding defined in Eq. S-1, precision and F-score increase by 0.4% and 0.2% respectively.

Analysis for G2AFF. The visualization of weight learning in our geometry-guided adaptive feature fusion is presented in Figure S1. In *Sample 1*, the highest weight is assigned to the view with the largest standard deviations of attention weight and relative pose distance. In *Sample 2*, due to occlusion in *View 3* and *View 4*, the voxel weights in these two views are very low. *Sample 3* assigns a higher weight to the view with a larger standard deviation of relative pose distance, while *Sample 4* gives more attention to the view with a larger standard deviation of attention weight. In conclusion, our G2AFF is able to learn appropriate weights from the 3D geometry.

Additional qualitative results on ScanNet [2]. More visualizations on ScanNet are shown in Figure S2. Compared to SOTA methods (*i.e*. SimpleRecon [5], NeuralRecon [9], and VoRTX [7]), NeuralRecon + ours is able to reconstruct better meshes. Compared to VoRTX, VoRTX + ours can recall more regions and generate flatter meshes, *e.g*. for walls. It can be demonstrated that our geometry integration mechanism is helpful and can be plugged into both online (*e.g*. NeuralRecon) and offline (*e.g*. VoRTX) volumetric methods.

Qualitative results on 7-Scenes [6] and TUM RGB-D [8]. Qualitative results on 7-Scenes and TUM RGB-D datasets are given in Figure S3 and Figure S4. Compared to NeuralRecon and VoRTX, our proposed geometry integration mechanism can recall more meshes and reconstruct flatter planes, *e.g*. for the wall and floor.

References

[1] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13137–13146, 2021.

[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*, pages 5828–5839, 2017.

[3] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of European Conference on Computer Vision*, pages 405–421, 2020.

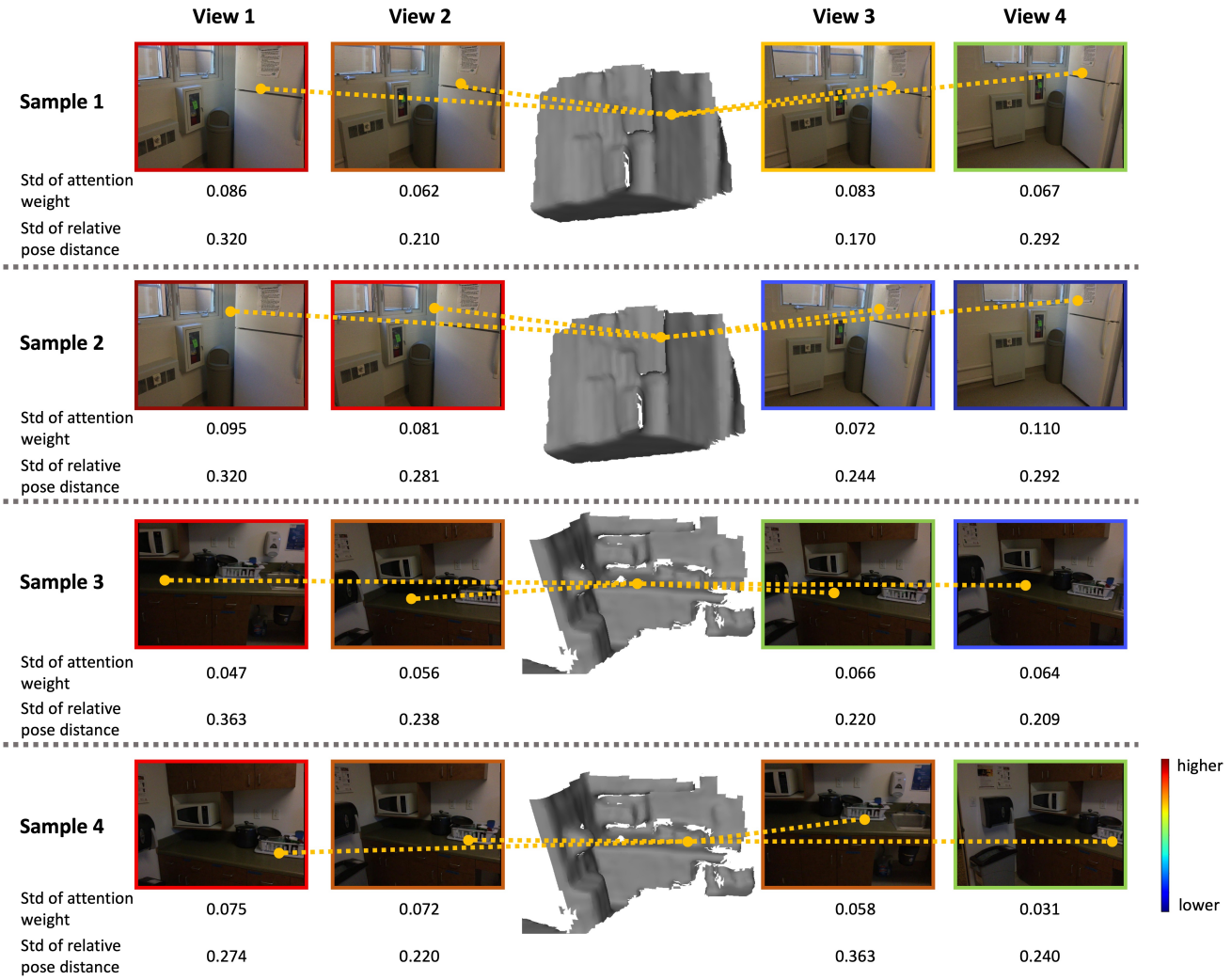

Figure S1. **Visualization for weight learning in our geometry-guided adaptive feature fusion.** The view weight of each sample decreases progressively from *View 1* to *View 4*, with the color of the image box as a clearer indication of the weight. The standard deviation (std) of attention weight and relative pose distance is also given. The yellow dots represent the sampled 3D voxels and their corresponding 2D pixels.

[4] Zak Murez, Tarrence Van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *Proceedings of European Conference on Computer Vision*, pages 414–431, 2020.

[5] Mohamed Sayed, John Gibson, Jamie Watson, Victor Prisacariu, Michael Firman, and Clément Godard. Simplerecon: 3d reconstruction without 3d convolutions. In *Proceedings of European Conference on Computer Vision*, pages 1–19, 2022.

[6] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937, 2013.

[7] Noah Stier, Alexander Rich, Pradeep Sen, and Tobias Höllerer. Vortx: Volumetric 3d reconstruction with transformers for voxelwise view selection and fusion. In *2021 International Conference on 3D Vision*, pages 320–330, 2021.

[8] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proceedings of the International Conference on Intelligent Robot Systems*, pages 573–580, 2012.

[9] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021.

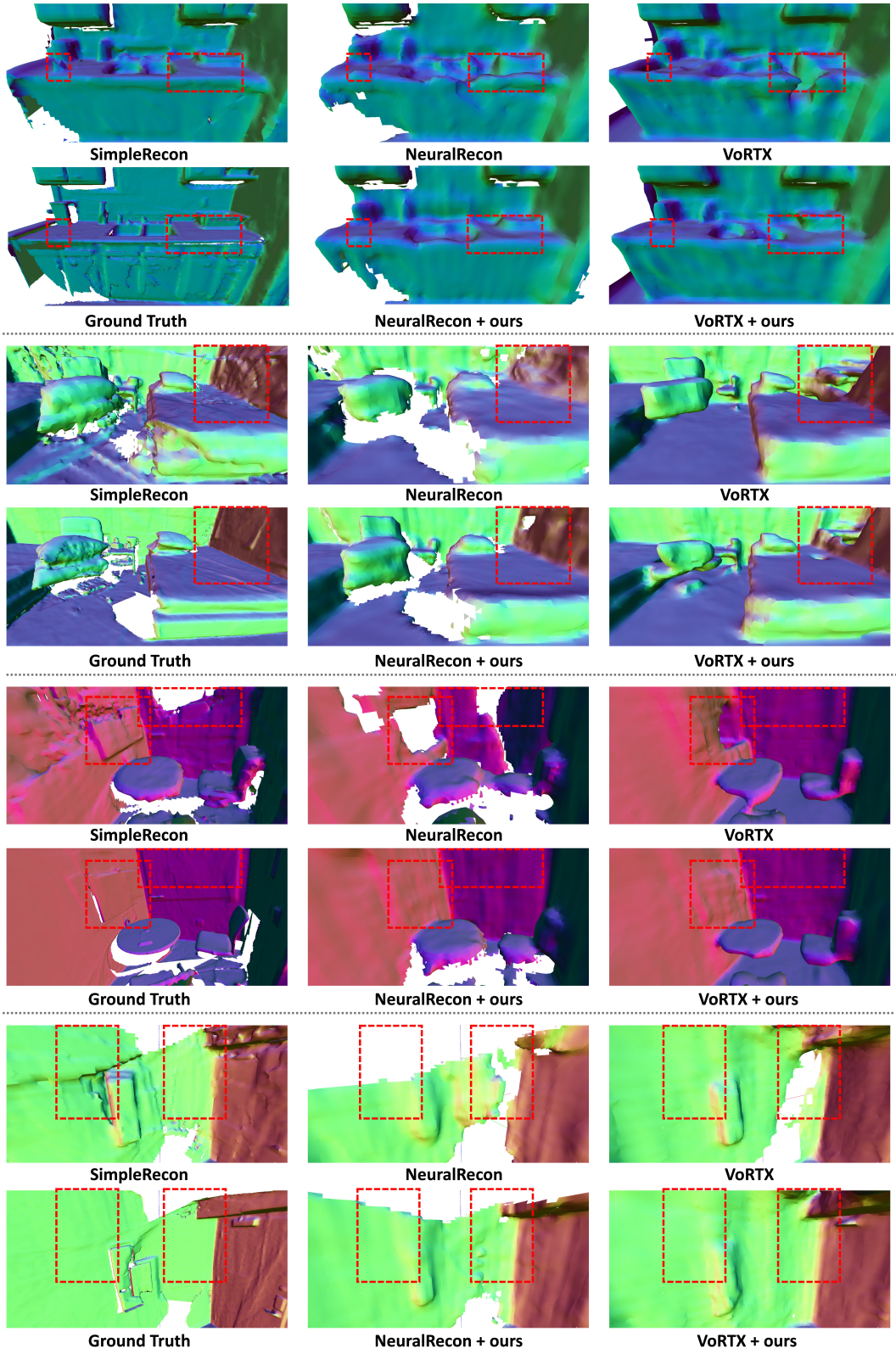

Figure S2. **Qualitative results on ScanNet.**

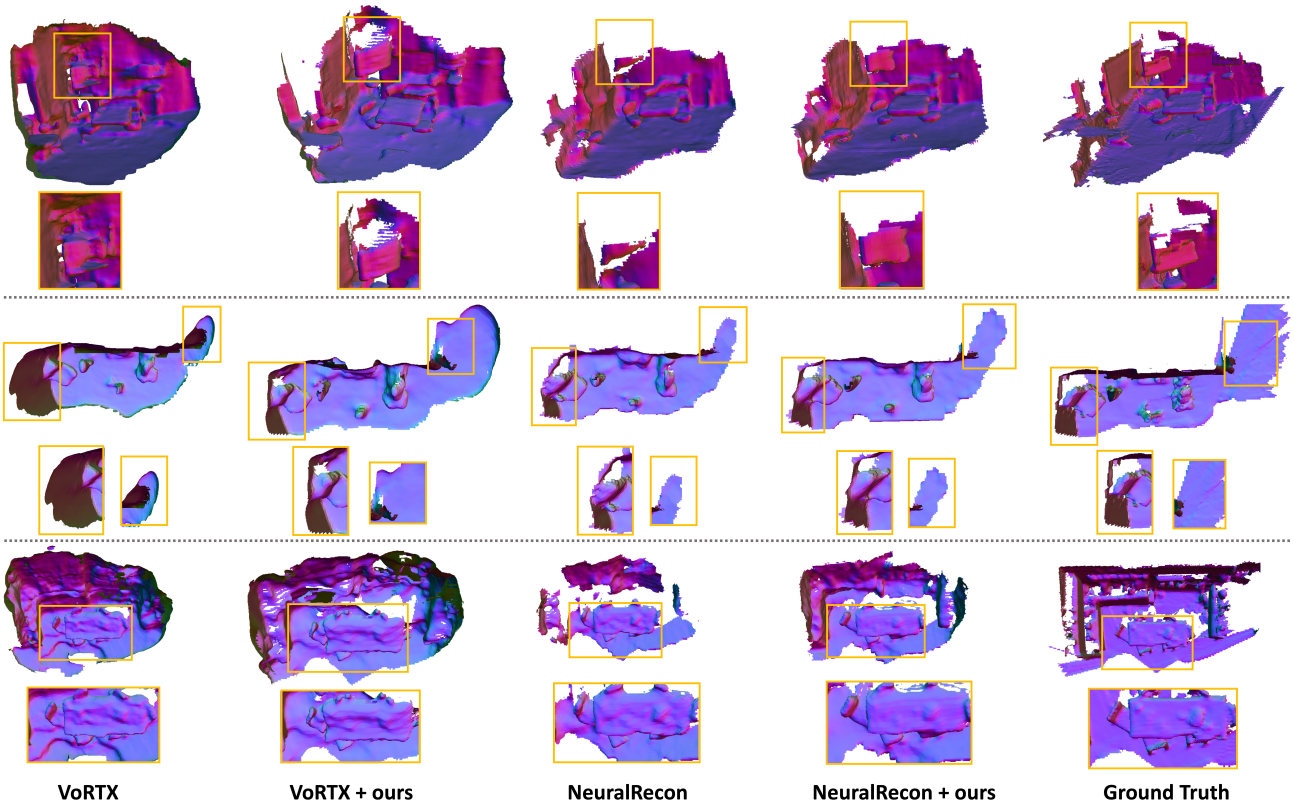

Figure S3. **Qualitative results on 7-Scenes.**

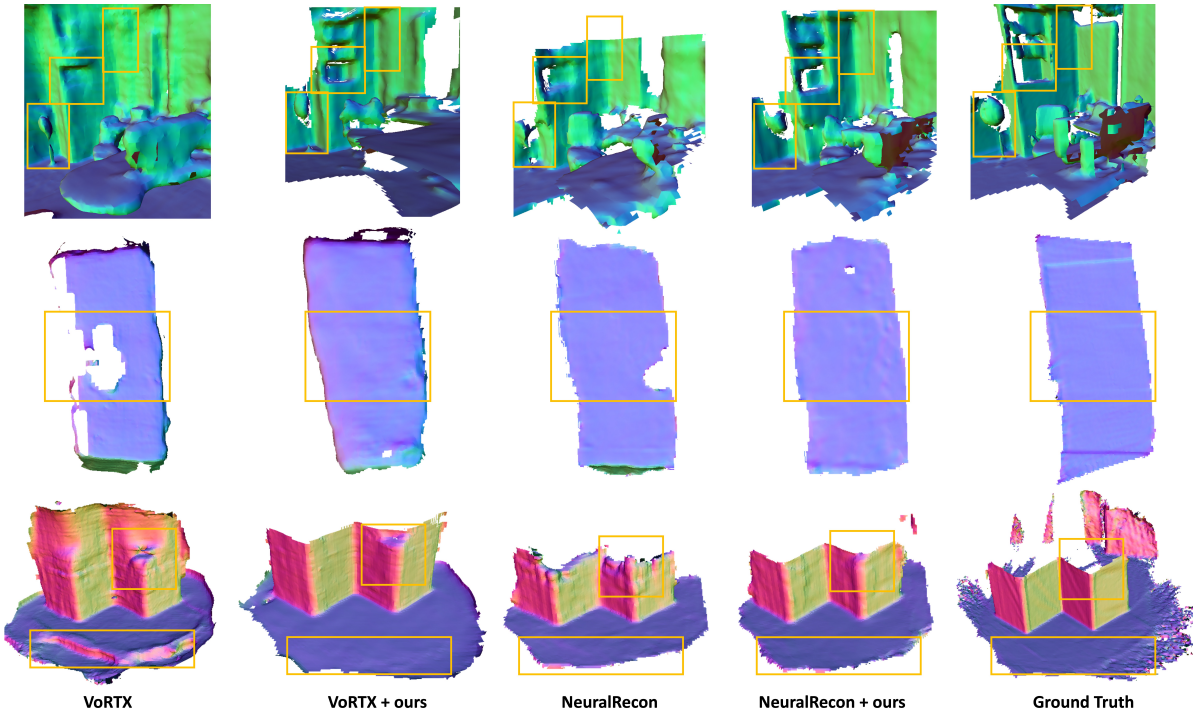

Figure S4. **Qualitative results on TUM RGB-D.**