Supplementary Materials of SurfelNeRF: Neural Surfel Radiance Fields for Online Photorealistic Reconstruction of Indoor Scenes

We show more visualizations and comparisons in the supplementary and attached videos. Please watch the supplementary videos for more results. We also provide implementation details and visualization results of ablation studies. In particular, this supplementary material provides:

- Additional qualitative results and comparisons, as shown in attached videos.
- Implementation details about our proposed rasterization-guided render scheme and network architecture, as in Sec. 1.
- Additional ablations, including taking RGB images as input only (without sensor depth), and the comparison with our depth refinement module, as in Sec. 2.1.
- More qualitative results of ablation studies to validate the effectiveness of our module and provide further analyses, as in Sec. 2.2.

1. Implementation Details

1.1. Network Details

Image feature extractor. We follow [8] to use the same modified MnasNet [4] pretrained from ImageNet as a 2D CNN to extract surfel features from images. For each surfel, we extract multi-scale image features from corresponding pixels. The channel number of extracted image features is 83. We then project extracted image features to surfel features with channel number of 32 by an MLP.

GRU fusion network. We employ a one layer GRU network to fuse the surfel features. Given features $\mathbf{f}_t^{\texttt{merge}}$ of input surfels and features $\mathbf{f}_{t-1}^{\texttt{corrs}}$ of corresponding global surfels, the process of updating global surfel features with GRU can be given as:

$$\mathbf{f}_t^{\texttt{corrs}} = \texttt{GRU}(\mathbf{f}_t^{\texttt{merge}}, \mathbf{f}_{t-1}^{\texttt{corrs}}), \tag{1}$$

where the detail is expresses by

$$\begin{aligned} \mathbf{z}_t &= M_z([\mathbf{f}_t^{\texttt{merge}}, \mathbf{f}_{t-1}^{\texttt{corrs}}]), \\ \mathbf{r}_t &= M_r([\mathbf{f}_t^{\texttt{merge}}, \mathbf{f}_{t-1}^{\texttt{corrs}}]), \\ \tilde{\mathbf{f}}_t^{\texttt{corrs}} &= M_t([\mathbf{r}_t * \mathbf{f}_{t-1}^{\texttt{corrs}}, \mathbf{f}_t^{\texttt{merge}}]), \\ \mathbf{f}_t^{\texttt{corrs}} &= (1 - \mathbf{z}_t) * \mathbf{f}_{t-1}^{\texttt{corrs}} + \mathbf{z}_t * \tilde{\mathbf{f}}_t^{\texttt{corrs}}, \end{aligned} \tag{2}$$

where M_z, M_r and M_t both have one MLP layer followed by a sigmoid, sidmoid and tanh activation function, respectively. $[\cdot, \cdot]$ denotes the operation of concatenate.

Rendering module. We employ an MLP-like rendering module, $\texttt{Render}(x_i, \mathbf{f}^i(\mathbf{x}_i), \mathbf{d})$, to predict volume density σ_i and radiance c_i at each shading point x_i with giving "interpolated" surfel features $\mathbf{f}^i(\mathbf{x}_i)$ and its view direction $\mathbf{d}$. Specifically, the details can be given as

$$\begin{aligned} \sigma_i &= F_\sigma([\mathbf{f}^i(\mathbf{x}_i), \gamma(\mathbf{x}_i)]), \\ c_i &= \text{Sigmoid}(F_r([\mathbf{f}^i(\mathbf{x}_i), \gamma(\mathbf{d})])), \end{aligned} \tag{3}$$

where $F_\sigma(\cdot)$ is a one layer MLP network with the ReLU activation function. $F_r(\cdot)$ is an MLP network with four layers and ReLU activation function, where the channel number of all hidden layers is 256. $\gamma(\cdot)$ denotes the positional embedding with maximum frequency of 5. $\mathbf{x}_i$ is the position of shading point. $[\cdot, \cdot]$ is the concatenation operation. The "interpolated" surfel features $\mathbf{f}^i(\mathbf{x}_i)$ are obtained based on the intersection of ray and surfels as

$$\mathbf{f}^i(\mathbf{x}_i) = \frac{r^i - \|\mathbf{x}^i - \mathbf{p}^i\|}{r^i} \texttt{F}(\mathbf{f}^i, \mathbf{d}, \mathbf{n}^i, w^i), \tag{4}$$

where $\mathbf{p}^i$, $\mathbf{f}^i$, $\mathbf{n}^i$, r^i and w^i indicate the position, features, normal, radius and weight of surfels s^i respectively. The function $\texttt{F}$ is a MLP-like network, which is given as

$$\begin{aligned} &\texttt{F}(\mathbf{f}^i, \mathbf{d}, \mathbf{n}^i, w^i) = \\ &\quad F_f([\mathbf{f}^i, \gamma(\mathbf{d}), \gamma(w^i), \gamma(\mathbf{n}^i), \gamma(\mathbf{d} - \mathbf{n})]), \end{aligned} \tag{5}$$

where F_f is a two layer MLP network with ReLU activation function and the channel number of hidden layers is 256.

Methods	PSNR↑	SSIM↑	LPIPS ↓	Time ↓
Instant-NGP [1]	23.23	0.714	0.459	0.03s
ADOP [2]	25.01	0.807	0.272	1s
NeRFingMVS [6]	26.37	0.903	0.245	-
IBRNet [5]	25.14	0.871	0.266	-
NeRFusion [8]	26.49	0.915	**0.209**	38s
PointNeRF [7]	28.99	0.829	0.324	30s
SurfelNeRF	29.58	0.919	0.215	0.2s
SurfelNeRF (MVS)	**29.74**	**0.920**	0.211	0.2s

Table 1. Quantitative comparisons with SOTAs on the ScanNet dataset with per-scene optimization. SurfelNeRF (MVS) indicates taking RGB input with estimated depth via the MVS depth estimator. Time↓ indicate average time to render an image.

Methods	PSNR↑	SSIM↑	LPIPS ↓
IBRNet [5]	21.19	0.786	0.358
NeRFusion [8]	22.99	0.838	0.335
PointNeRF [7]	20.47	0.642	0.544
SurfelNeRF	23.82	0.845	0.327
SurfelNeRF (MVS)	**24.29**	**0.871**	**0.324**

Table 2. Quantitative comparisons with SOTAs on the ScanNet dataset with no per-scene optimization. SurfelNeRF (MVS) indicates taking RGB input with estimated depth via the MVS depth estimator.

Overall, the function F takes features of surfels and corresponding geometry attributes of surfels and rays as input, and outputs view-dependent surfel features. The view-dependent surfel features are then weighted based on the radius and the distance between intersections and centers of surfels. The far the intersections are, the less they contribute to the interpolated features.

2. Additional Results

In this section, we report and analyse quantitative and qualitative results of additional ablation studies, and provide additional qualitative comparison with recent SOTA methods.

2.1. Depth from MVS

We conduct an additional experiment that takes the RGB input from sensors only and employ a off-the-shelf depth estimator [3] to obtain estimated MVS depth maps. We show the quantitative results of this setting with direct network inference and per-scene optimization in Table. 2 and Table. 1, respectively. Our method with estimated depth, called SurfelNeRF(MVS), achieve comparable performance with using sensor depth and a depth refinement network. The slight improvement comes from the scenes where depth captured from the sensor appears heavily incomplete and noisy, where the off-the-shelf depth estimator [3] can produce better depth via multi-view stereo than raw sensor measurements. To investigate the influence of depth quality, we conduct an additional ablation study about depth refinement network in the next section.

Fusion Scheme		PSNR↑	SSIM↑	LPIPS ↓
Weighted Sum	No per-scene	23.09	0.833	0.353
GRU	optimization	23.82	0.845	0.327
Weighted Sum	Per-scene	28.54	0.884	0.293
GRU	optimization	**29.58**	**0.919**	0.215

Table 3. Ablation studies about fusion schemes in our SurfelNeRF.

2.2. Additional Ablation Studies

Depth refinement network. We show the visualization results of heavily incomplete scenes with different input depth, including depth captured from sensors, refined from the depth refinement network, and estimated by the off-the-shelf depth estimator, as shown in Figure. 1. For the first column that a novel view from $scene0000\_01$ in ScanNet, the sensor cannot capture the high-quality depth around the thin bicycle wheels and the far away television. With the help of RGB input or multi-view stereo techniques, the depth refinement network and depth estimator can fill the depth, which reconstruct surfels better and providing better rendering results. Comparing with the depth refinement network, the off-the-shelf depth estimator produce higher quality of estimated depth since it spends extra time to consider the prior of multi-view stereo. Comparing the results with different depth quality, this results shows that the higher quality of depth the better photo-realistic rendering results since depth quality decides the quality of reconstruction surfels.

Fusion scheme. To investigate the effectiveness of the GRU fusion module, we have conducted an ablation study and shown quantitative results in the main paper. We recap the results which is shown in Table. 3 and provide the qualitative results in Figure. 2.

References

[1] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. 2

[2] Darius Rückert, Linus Franke, and Marc Stamminger. Adop: Approximate differentiable one-pixel point rendering. *ACM Transactions on Graphics (TOG)*, 41(4):1–14, 2022. 2

[3] Mohamed Sayed, John Gibson, Jamie Watson, Victor Prisacariu, Michael Firman, and Clément Godard. Simplere-

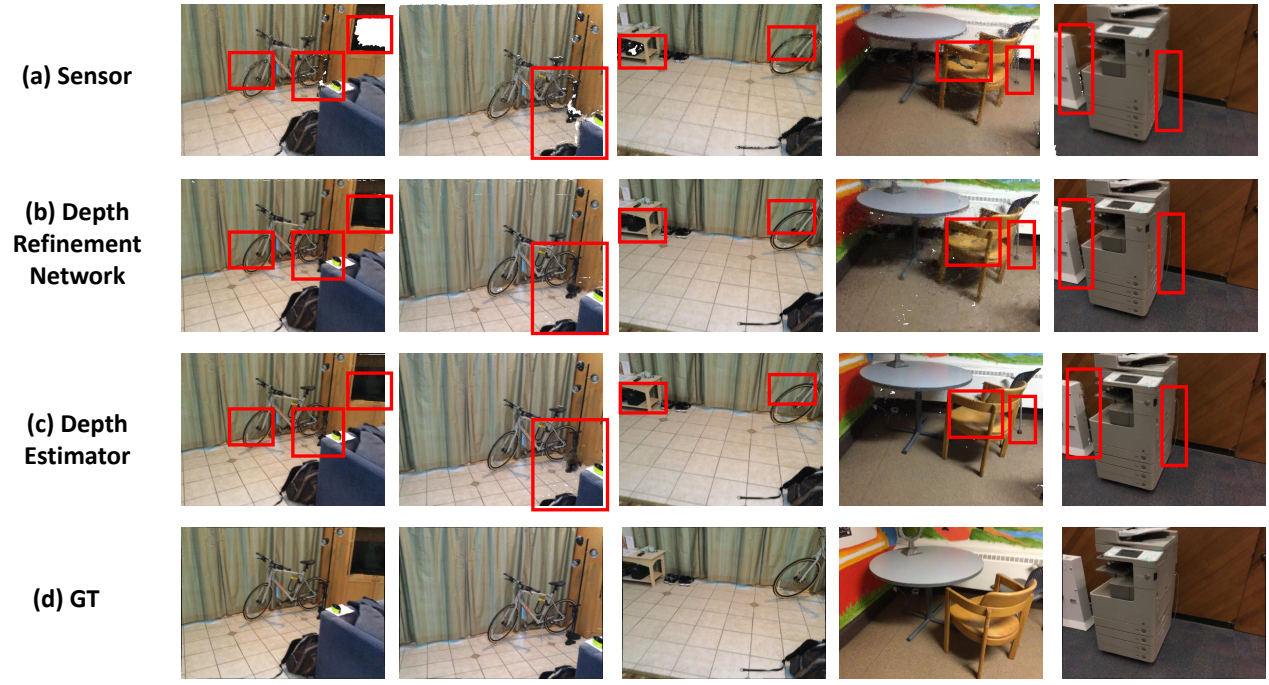

Figure 1. Comparison of different types of input depth on the per-scene optimization setting. Per-sceen optimizaiton would not change the surfel position and number, so it can obtain the same conclusion when evaluating on the no per-scene optimization setting. The highlight areas are indicated by red rectangles. It is obvious that depths captured from sensor may be incomplete and noisy, which affects the surfel reconstruction resulting in sub-optimal rendering results.

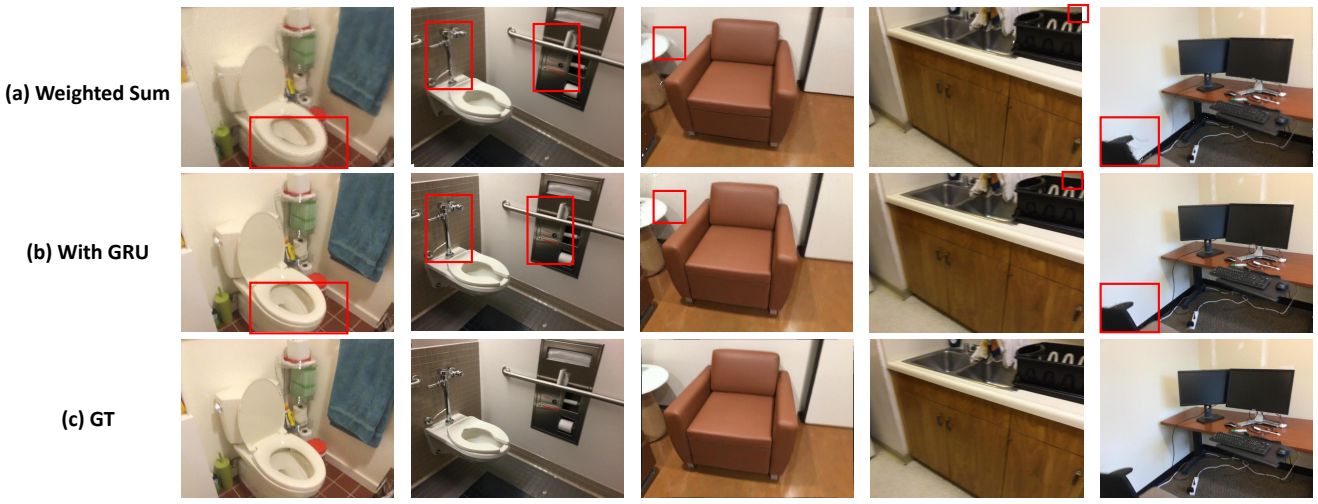

Figure 2. Comparison of different fusion schemes on the per-scene optimization setting. The highlight areas are indicated by red rectangles. As can be seen in the figure, GRU can generate sharper and clearer details in novel view synthesis. GRU has the capability to adaptively update features based on high-level features, which makes the fusion process more robust.

con: 3d reconstruction without 3d convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2

[4] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2820–2828, 2019. 1

[5] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-

Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 2

[6] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5610–5619, 2021. 2

[7] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022. 2

[8] Xiaoshuai Zhang, Sai Bi, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Nerfusion: Fusing radiance fields for large-scale scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5449–5458, 2022. 1, 2