—Supplementary Material—
Neural Inverse Rendering from Propagating Light

Anagh Malik*[1,2] Benjamin Attal*[3] Andrew Xie[1,2] Matthew O'Toole+[3] David B. Lindell+[1,2]
[1]University of Toronto [2]Vector Institute [3]Carnegie Mellon University
*joint first authors +equal contribution
https:/anaghmalik.com/InvProp

We provide additional implementation details related to the architecture of our model, optimization procedure, and experimental settings. We also include supplemental results and details about the captured dataset. Code and data are available from our project webpage. **Please refer to the webpage and video** for animated visualizations of results, including lidar view synthesis, reconstructed geometry, time-resolved relighting, and separation of direct and indirect light.

1. Implementation Details

1.1. Architecture Details

Geometry. We use Zip-NeRF's [3] proposal sampling architecture to represent scene geometry and for volume rendering. Specifically, we use two hash-encoding-based "proposal" networks that output density, which is used for hierarchical sampling, and one final network that outputs the density used in Equation 4, as well as normals $\mathbf{n}$ used for the cache and physically-based rendering. The hash-encoding based proposal networks have spatial resolutions of 512 and 1024 along all axes, while the final density network has a resolution of 2048. Each network has a multi-layer perception (MLP) head with 2 layers and 64 hidden units.

We use 64 samples for the first proposal network, 64 samples for the second proposal network, and 32 samples for the final geometry network to volume render the cache geometry. In order to render the physically-based model, we leverage a single sample quadrature estimator for both primary and secondary rays, as in Attal et al. [2].

Cache. The position-dependent appearance feature $\mathbf{f}^{app}$ used for the cache has dimension 128 and is predicted with a hash encoding that has a spatial resolution of 2048. The learned BRDF for the direct component of the cache f^{dir} in Equation 11 is a sum of diffuse BRDF $f^{dir,diff}(\mathbf{f}^{app})$, and specular BRDF $f^{dir,spec}(\mathbf{f}^{app}, \mathbf{n}, \boldsymbol{\omega}_\ell, \boldsymbol{\omega}'_\mathrm{o})$. We predict the diffuse BRDF as a function of $\mathbf{f}^{app}$ alone, with a 2-layer, 64-hidden-unit MLP. We predict the specular BRDF as a function of $\mathbf{f}^{app}$ as well as the dot product between the normal $\mathbf{n}$ and normalized half vector $\frac{\boldsymbol{\omega}_\ell+\boldsymbol{\omega}'_\mathrm{o}}{||\boldsymbol{\omega}_\ell+\boldsymbol{\omega}'_\mathrm{o}||}$ with a 2-layer, 64-hidden-unit MLP.

The specular indirect component of the cache, as described in Equation 12, uses a split-sum approximation. We predict $f_\Omega^{indir}(\mathbf{f}^{app}, \mathbf{n}, \boldsymbol{\omega}'_\mathrm{o})$ as a function of the appearance feature and the dot product between normals $\mathbf{n}$ and outgoing direction $\boldsymbol{\omega}'_\mathrm{o}$. We predict $L_{\mathrm{i},\Omega}^{indir}(\mathbf{f}^{app}, \mathbf{x}_\ell, \mathbf{n}, \boldsymbol{\omega}'_\mathrm{o})$ as a function of the appearance feature, the reflected direction $\mathit{reflect}(\boldsymbol{\omega}'_\mathrm{o}, \mathbf{n})$, and the light source position $\mathbf{x}_\ell$. Again, both use 2-layer, 64 hidden unit MLPs. We also predict a purely diffuse indirect component $L_\mathrm{o}^{indir,diff}$ that is a function of the appearance feature and is conditioned on light source position, with a 2-layer 64-hidden-unit MLP.

Materials. We leverage the Disney–GGX [4] BRDF parameterization, with parameters albedo $\mathbf{a}(\mathbf{x})$, metalness $m(\mathbf{x})$, and roughness $r(\mathbf{x})$. This BRDF can be written as:

$$f(\boldsymbol{\omega}_\mathrm{i}, \boldsymbol{\omega}_\mathrm{o}, \mathbf{x}) = f_{diffuse}(\mathbf{x}) + f_{specular}(\boldsymbol{\omega}_\mathrm{i}, \boldsymbol{\omega}_\mathrm{o}, \mathbf{x}) \quad \text{(S1)}$$

$$f_{diffuse}(\mathbf{x}) = \frac{(1-m(\mathbf{x}))\mathbf{a}(\mathbf{x})}{\pi} \quad \text{(S2)}$$

$$f_{specular}(\boldsymbol{\omega}_\mathrm{i}, \boldsymbol{\omega}_\mathrm{o}, \mathbf{x}) = \frac{DFG}{4(\mathbf{n}\cdot\boldsymbol{\omega}_\mathrm{i})(\mathbf{n}\cdot\boldsymbol{\omega}_\mathrm{o})} \quad \text{(S3)}$$

We refer to Burley [4] and Liu *et al.* [6] for definitions of (D, F, G). We use the Trowbridge-Reitz distribution function [10] for the normal distribution function D.

We predict a material feature $\mathbf{f}_{mat}$ using a hash-encoding-based network with a resolution of 2048, and decode all of the above parameters using a linear layer from this feature.

Importance sampling. We leverage multiple importance sampling (MIS) [10], using the distribution function of the GGX BRDF, and a learned von Mises-Fisher-based importance sampler with an architecture similar to that of Attal

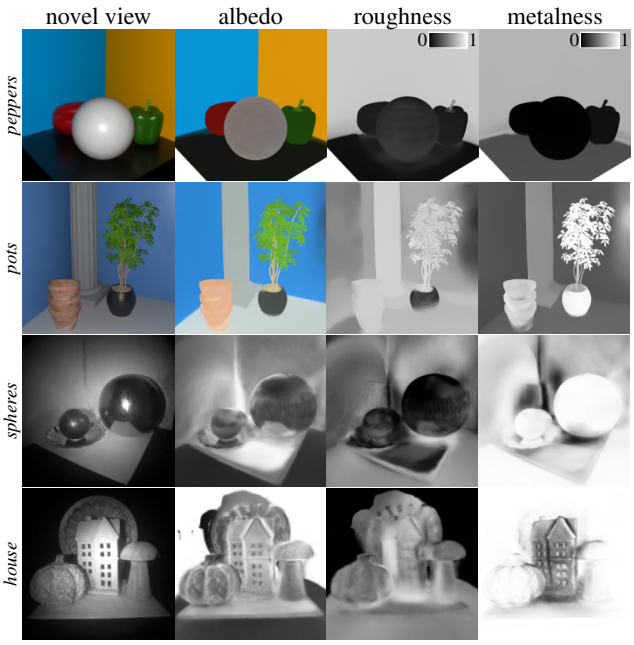

Fig. S1. Rendered views and materials for simulated (rows 1–2) and captured scenes (rows 3–4). See the text for a detailed description.

et al. [2]. We supervise the importance sampler using the integrated intensity along secondary rays.

1.2. Loss Details

Mask loss. The mask loss for a particular ray is defined as:

$$\mathcal{L}_{mask} = mask \cdot |1 - acc| + (1 - mask) \cdot |acc|, \tag{S4}$$

where *acc* is the accumulated transmittance (or sum of the render weights) along a particular ray.

Predicted normal loss. As discussed, we output the predicted normals using the density hash-encoding-based network. Similar to Ref-NeRF [12] and TensoIR [5], we constrain the predicted normals to match the negative gradient of the density field with an L2 loss:

$$\mathcal{L}_{normals} = \sum_k w_k \left\| \mathbf{n}_k^{pred} - \mathbf{n}_k^{derived} \right\|^2, \tag{S5}$$

where w_k are the render weights for a given ray, and

$$\mathbf{n}_k^{derived} = -\frac{\nabla\sigma(\mathbf{x}_k)}{\|\nabla\sigma(\mathbf{x}_k)\|}. \tag{S6}$$

The loss weight $\lambda_{normals}$ varies per-dataset.

Material smoothness loss. For the smoothness los $\mathcal{L}_{mat}$, we leverage the implementation of TensoIR [5] for synthetic datasets, and a standard L2 smoothness loss for captured datasets.

RawNeRF loss. For the photometric losses (Equations 13 and 14 of the main paper), we use $\beta = 1$ for synthetic scenes, $\beta = 2$ for the cache in captured scenes, and $\beta = 1$ for the physically-based model in captured scenes.

Other loss hyperparameters. For our photometric losses, we set $\lambda_{cache} = 10$, $\lambda_{dir} = 1$, $\lambda_{indir} = 1$. For the additional losses, we set $\lambda_{interlevel} = 0.01$ for all scenes. For the simulated scenes, we set $\lambda_{geom} = 0.0008$, $\lambda_{disortion} = 0.0001$, and $\lambda_{mask} = 0.1$. For the captured scenes, we set $\mathcal{L}_{geom} = 0.00025$ and $\lambda_{disortion} = 0.001$. We assume that the scene mask is all ones (i.e., all opaque) for captured scenes, and we set the mask loss to $\lambda_{mask} = 0.001$.

1.3. Time-Resolved Imaging Without Lidar

Section 5.3 of the paper discusses how our model can recover time-resolved videos of propagating light by training on indirect time-of-flight or intensity images. In both cases, we write the loss as

$$\mathcal{L}_{data} = \sum_{\mathbf{o},\boldsymbol{\omega}_o} \alpha(L_{\mathrm{i}}^{cache}) \sum_k \left\| \sum_\tau g_k(\tau)(L_{\mathrm{i}} - L_{\mathrm{i}}^{meas}) \right\|^2. \tag{S7}$$

Here, $\{g_k(\cdot)\}_k$ defines a set of path length importance functions induced by the indirect time-of-flight or intensity sensor [1]. For indirect time-of-flight, we have:

$$g_k(\tau) = \cos(2\pi f_k \cdot \tau + \theta_k) + 1, \tag{S8}$$

where f_k are frequencies and θ_k are phase shifts. We use $(f_1, \theta_1) = (30 \times 10^6, 0)$, $(f_2, \theta_2) = (30 \times 10^6, \pi)$, $(f_3, \theta_3) = (170 \times 10^6, 0)$, $(f_4, \theta_4) = (170 \times 10^6, \pi)$. For intensity images, we use $g_1 = 1$. We apply the same consistency loss as in Equation 15 of the main paper without adjustments.

1.4. Finetuning for Relighting

As discussed in Section 5.3 of the paper, we leverage finetuning for relighting whenever the intensity profile of the light source differs from the training data (e.g. a projector as in Fig. 1 of the paper). In order to do this, we freeze all model parameters, apart from those that define the cache direct and indirect appearance (Equation 11 and Equation 12). We then train these parameters in order to minimize the radiometric prior (Equation 15).

Table S1. Evaluation of lidar rendering from novel viewpoints and geometry recovery.

	method	PSNR (dB) ↑	LPIPS ↓	SSIM ↑	MAE ↓	L1 depth ↓	T-IOU ↑
sim	T-NeRF [7]	22.44	0.40	0.71	28.00	0.59	0.58
	T-NeRF w/ filtering	24.52	0.34	0.78	22.54	0.40	0.70
	FWP++ [8]	29.00	0.30	0.87	22.80	0.47	0.73
	ours	30.99	0.31	0.89	8.45	0.21	0.76

2. Additional Results

2.1. Material Decomposition

In Fig. S1, we show the recovered albedo, roughness, and metalness for simulated and captured scenes from a novel view. Qualitatively, the results align with expectations in several respects. The recovered albedo factors out variations in shading and illumination; the roughness is low/dark for specular objects (floor, ball, peppers in row 1; pot in row 2; chrome balls in row 3); and the metalness is bright/high for the pot in row 2 and chrome balls in row 3. Generally, the materials are harder to interpret for the captured results—though we expect that improvements to the system calibration would likely improve the results.

We note that for Fig. S1, we leverage an additional loss applied to the *integrated* time-resolved measurements — specifically the loss in Equation S7 for intensity images. We find that this slightly improves the convergence of the recovered materials.

2.2. Additional Baselines

We include another T-NeRF [7] baseline, which applies a matched filter to the time-resolved measurement to find the direct peak—similar to a conventional lidar—before supervision. We include this result in Table S1 (see T-NeRF w/ filtering).

The baseline improves upon T-NeRF and even outperforms FWP++ for geometry modeling. This is expected since one of the main reasons T-NeRF fails in geometry recovery is the presence of the indirect component of light in the lidar scans. However, our method still outperforms this new baseline since the matched filter does not always accurately localize the time of the direct surface reflection, especially under strong indirect light. The new baseline also struggles with novel view synthesis since it does not model indirect light transport effects.

2.3. Quantitative Results

We provide a per-scene breakdown of quantitative results for simulated scenes in Table S2 and captured scenes in Table S3. We see similar trends for all scenes as described in the main text.

2.4. Qualitative Results

We provide additional qualitative results on novel views in Figure S3 and in the supplemental web page, which includes novel view flythroughs, time-resolved relighting, and separation of direct and indirect light. We emphasize that our method recovers more accurate geometry, particularly in scenarios involving strong indirect lighting from specular reflections or diffuse inter-reflections, outperforming previous approaches.

3. Dataset

3.1. Calibration

To capture our real multi-viewpoint dataset, we use a hardware setup similar to the one used by Malik et al. [8], with a 532 nm laser emitting 35 ps pulses at a 10 MHz synced with a single pixel scanning SPAD at 512×512 resolution. We capture multiple viewpoints with the same rotation table and elevation arm setup. Specifically, our light source position is fixed for all viewpoints with respect to the camera rather than to the scene. Camera intrinsics are calibrated with a checkerboard and the MATLAB Camera Calibration Toolbox [9], and extrinsics are calibrated using COLMAP [11] with a scene including a checkerboard so that radial camera pose translation can be scaled by matching the reconstruction to the board's known geometry.

For our scenes, we assume our light sources are point sources, calibrated so that their location is known with respect to the scene. We simulate point light sources by passing our free-space laser light, coupled through multi-mode fiber, through a collimating lens, and multiple high-power diffusers. To address any residual imperfections in our non-ideal point source, we image a uniformly reflective, diffuse surface with a pre-calibrated pose, using a checkerboard pattern for alignment. This process enables us to compute a directional intensity profile for the light source, which we model during inverse rendering.

The light source position is calibrated using the following procedure. We (1) capture a checkerboard and compute corner poses, (2) use the corresponding time-resolved measurement for each corner to measure total ToF and thus distance from the light source to the camera, (3) subtract the calibrated corner pose to camera distance, and (4) trilaterate to locate the unknown light source position.

3.2. Scene Descriptions

We provide a description of each captured scene in Table S4.

Table S2. Breakdown of results on the simulated scenes for PSNR, LPIPS, SSIM, MAE, L1 Depth (L1) and Transient IOU (T-IOU).

		Pots	Cornell	Peppers	Kitchen
PSNR	T-NeRF	23.78	23.90	19.07	23.00
	FWP	28.64	31.75	33.01	22.61
	ours	30.44	32.38	37.46	23.68
LPIPS	T-NeRF	0.36	0.32	0.44	0.49
	FWP	0.26	0.30	0.26	0.39
	ours	0.35	0.31	0.27	0.30
SSIM	T-NeRF	0.73	0.82	0.72	0.56
	FWP	0.86	0.87	0.94	0.79
	ours	0.90	0.89	0.93	0.84
MAE	T-NeRF	36.09	18.33	13.03	44.56
	FWP	37.41	10.86	7.20	35.75
	ours	7.81	10.25	2.65	13.08
L1	T-NeRF	0.18	0.10	0.42	1.66
	FWP	0.29	0.10	0.28	1.20
	ours	0.04	0.09	0.19	0.53
T-IOU	T-NeRF	0.66	0.69	0.76	0.20
	FWP	0.82	0.82	0.88	0.41
	ours	0.88	0.78	0.94	0.46

Table S3. Breakdown of results on the captured scenes for PSNR, LPIPS, SSIM, MAE and Transient IOU (T-IOU).

		House	Globe	Spheres	Statue
PSNR	T-NeRF	15.94	11.44	13.25	18.05
	FWP	27.40	26.00	28.51	31.89
	ours	27.47	25.97	26.07	30.04
LPIPS	T-NeRF	0.46	0.56	0.53	0.58
	FWP	0.30	0.34	0.38	0.26
	ours	0.32	0.34	0.39	0.28
SSIM	T-NeRF	0.36	0.19	0.35	0.51
	FWP	0.78	0.75	0.81	0.92
	ours	0.79	0.75	0.75	0.90
T-IOU	T-NeRF	0.34	0.10	0.13	0.34
	FWP	0.62	0.54	0.43	0.60
	ours	0.60	0.53	0.44	0.60

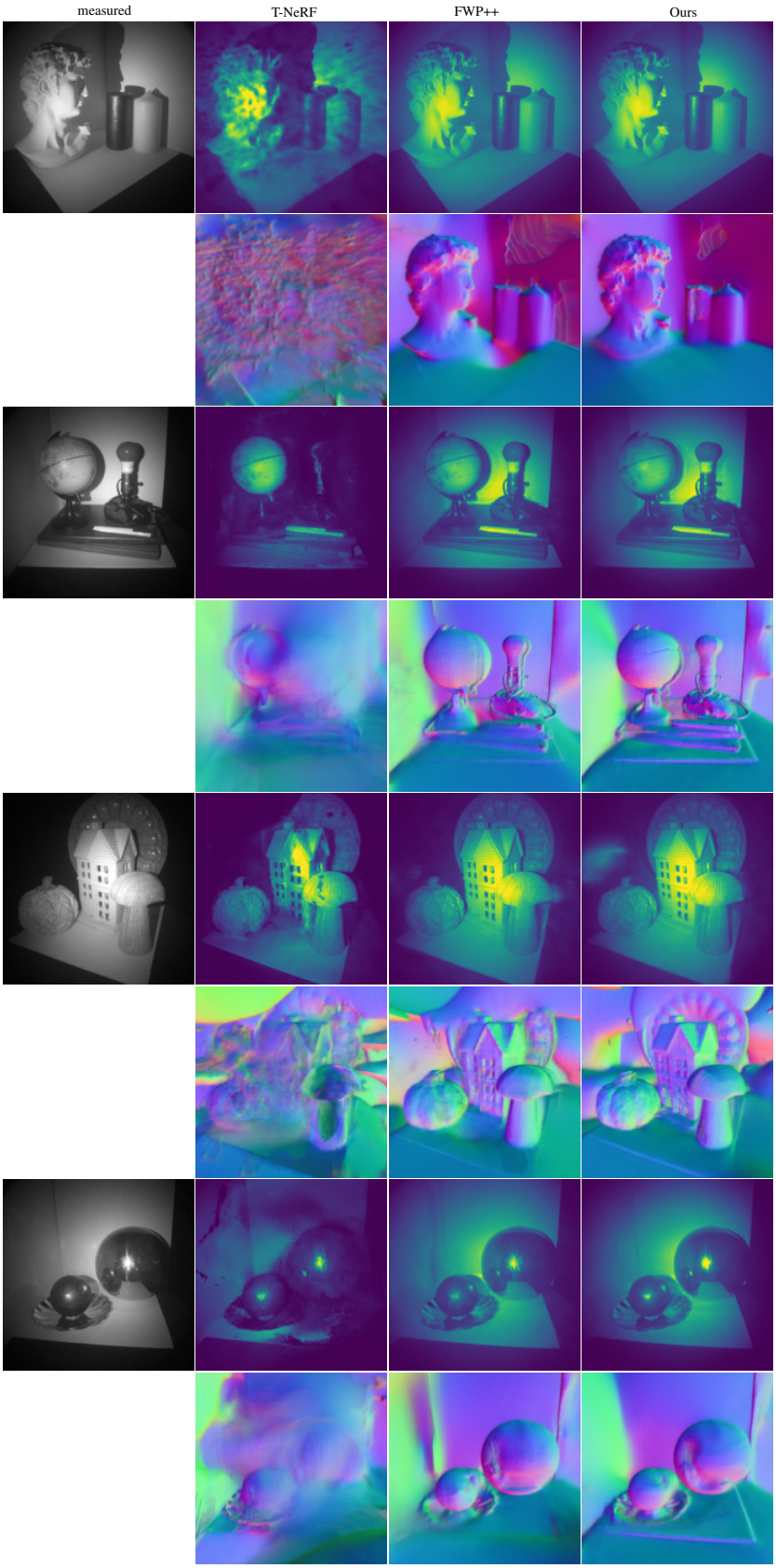

Fig. S2. Additional captured results comparing reconstructed normals from the proposed method to those of T-NeRF [7] and FWP++ [8].

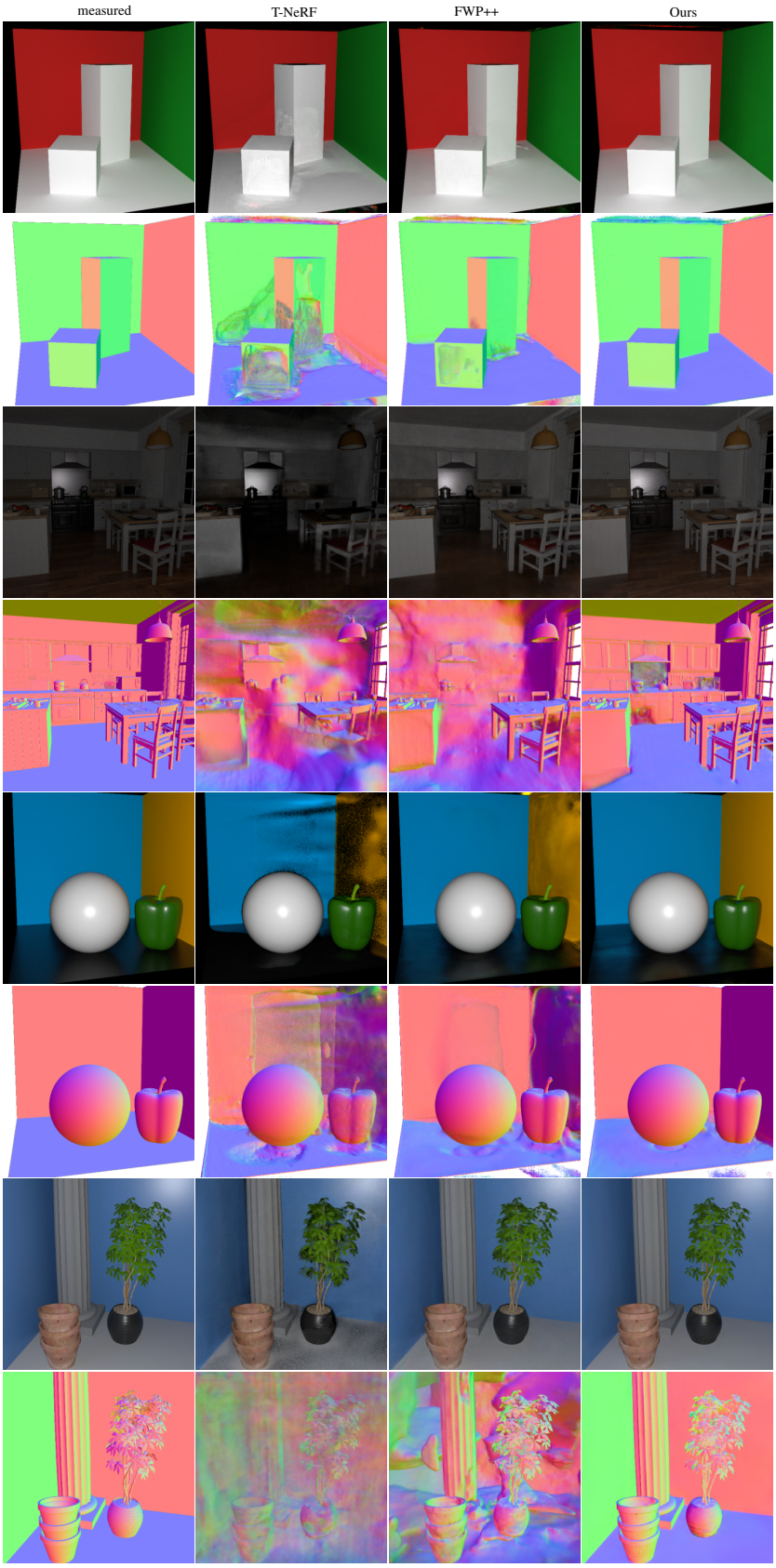

Fig. S3. Additional simulated results comparing rendered novel views and reconstructed normals from the proposed method to those of T-NeRF [7] and FWP++ [8].

Table S4. Descriptions of the captured scenes. All scenes have a calibrated bin width of 0.0105 m and span 15 degrees in elevation angle.

Scene Description	**Description**	**Training Views**	**Test Views**	**Azimuth Span**	**Normalization Scale**
House	A diffused pulsed laser source rotates with the lidar sensor and illuminates a ceramic house, mushroom, and pumpkin with a plate in the background.	81	13	240°	600
Globe	A diffused pulsed laser source rotates with the lidar sensor and illuminates a globe and a lightbulb. Our model reconstructs the fine details of the wires of the lightbulb stand.	55	11	132°	600
Spheres	A diffused pulsed laser source rotates with the lidar sensor and illuminates two specular spheres.	56	11	132°	600
Statue	From the Flying with Photons Dataset [8]: a stationary diffused pulsed laser source illuminates a statue of David and two candles from the side.	60	15	150°	600

References

[1] Benjamin Attal, Eliot Laidlaw, Aaron Gokaslan, Changil Kim, Christian Richardt, James Tompkin, and Matthew O'Toole. TöRF: Time-of-flight radiance fields for dynamic scene view synthesis. *Proc. NeurIPS*, 2021. 2

[2] Benjamin Attal, Dor Verbin, Ben Mildenhall, Peter Hedman, Jonathan T Barron, Matthew O'Toole, and Pratul P Srinivasan. Flash cache: Reducing bias in radiance cache based inverse rendering. In *Proc. ECCV*, 2024. 1, 2

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-NeRF: Anti-aliased grid-based neural radiance fields. In *Proc. ICCV*, 2023. 1

[4] Brent Burley and Walt Disney Animation Studios. Physically-based shading at Disney. In *ACM SIGGRAPH Courses*, 2012. 1

[5] Haian Jin, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su. TensoIR: Tensorial inverse rendering. In *Proc. CVPR*, 2023. 2

[6] Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. NERO: Neural geometry and BRDF reconstruction of reflective objects from multiview images. *ACM Trans. Graph.*, 42 (4):1–22, 2023. 1

[7] Anagh Malik, Parsa Mirdehghan, Sotiris Nousias, Kiriakos N Kutulakos, and David B Lindell. Transient neural radiance fields for lidar view synthesis and 3D reconstruction. In *Proc. NeurIPS*, 2023. 3, 5, 6

[8] Anagh Malik, Noah Juravsky, Ryan Po, Gordon Wetzstein, Kiriakos N. Kutulakos, and David B. Lindell. Flying with photons: Rendering novel views of propagating light. In *Proc. ECCV*, 2024. 3, 5, 6, 7

[9] Mathworks. Camera calibrator app. https://www.mathworks.com/help/vision/ref/cameracalibrator-app.html, 2020. 3

[10] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. MIT Press, 2023. 1

[11] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proc. CVPR*, 2016. 3

[12] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *Proc. CVPR*, 2022. 2