

## 1255 A Ablations

1256 We include ablations to better understand the impact of: (1) the  
 1257 proposed use of *shadow transients* for shadow mapping, (2) number  
 1258 of illumination points, (3) single-photon lidar’s temporal dimension,  
 1259 (4) light that has bounced *more than two times*, (5) modeling pulse  
 1260 shape, noise, and timing jitter, and (6) amount of training data. We  
 1261 include quantitative results for these ablations in text below and  
 1262 qualitative results in Fig. 12, Fig. 13, and Fig. 14.

### 1264 A.1 Shadow Mapping Approaches

1265 Our approach draws on the idea of shadow transients to map lidar  
 1266 measurements to binary shadow maps for each illumination source.  
 1267 Rather than only using the multiplexed measurement as input, we  
 1268 also provide predicted two-bounce ToF. To understand the benefit of  
 1269 this approach, we compare it to the naive approach of using only the  
 1270 multiplexed input with supervised learning. Since there is no cue for  
 1271 which shadow to generate given a multiplexed measurement, we try  
 1272 two approaches: (1) predict shadows for all illumination points in a  
 1273 forward pass, (2) condition on illumination point and predict only  
 1274 the corresponding shadow in a forward pass. We find that the model  
 1275 is unable to learn accurate shadows with either of these approaches.  
 1276 While our proposed method – which uses predicted two-bounce  
 1277 ToF as an additional input – yields 0.0186 MAE and 0.959 IoU (Fig.  
 1278 12, row 6), (1) yields 0.0984 and 0.788, respectively, and (2) yields  
 1279 0.0896 and 0.799, respectively. Qualitative results for (1) and (2) are  
 1280 shown in Fig. 12, rows 3 and 4, respectively. We also tried explicitly  
 1281 computing the shadow transients by taking the absolute difference  
 1282 of the predicted two-bounce ToF transient and measured transient,  
 1283 and using this as input, which, while still accurate, results in slightly  
 1284 worse performance (0.0231 MAE, 0.950 IoU), as shown in Fig. 12, row  
 1285 5. These results indicate that using shadow transient information  
 1286 – either implicitly or explicitly – is critical for accurate shadow  
 1287 mapping from multiplexed lidar measurements.

### 1289 A.2 Number of Illumination Points

1290 We vary the number of illumination points – using 4, 25, and 100 –  
 1291 and study its impact on depth estimation, specular surface segmenta-  
 1292 tion, and shadow mapping (a proxy for occluded 3D reconstruction;  
 1293 i.e., inaccurate shadows lead to poor reconstruction). Intuitively,  
 1294 more illumination points increases ambiguity – as there are more  
 1295 unknown correspondences of peaks to illumination points. However,  
 1296 we find that this only reduces performance for shadow mapping –  
 1297 more illumination points results in higher performance when es-  
 1298 timating either depth or specular surfaces. Results are shown in  
 1299 Fig. 11 of the main text. As more illumination points are added,  
 1300 there is more redundancy in depth information – since each illumi-  
 1301 nation points provides a depth cue for all points light reflects to.  
 1302 Similarly, more illumination points increases the odds of a specular  
 1303 cue becoming available – since specular cues depend on light from  
 1304 a specular surface eventually being reflected back to the sensor.  
 1305 However, increasing the number of illumination points presents a  
 1306 serious challenge for the model to separate shadows, resulting lower  
 1307 performance. Thus, we hypothesize there is a *pareto optimum* that  
 1308 exists for number of illumination points, which results in higher  
 1309 depth, specular surface, and shadow accuracy. In our work, this  
 1310

1312 optimum occurs at 25 illumination points. More work is needed to  
 1313 understand if the trend of depth improving with more illumination  
 1314 points continues with significantly more illumination points – on  
 1315 one hand, these additional illumination points provide redundancy  
 1316 in depth information, but, on the other, they increase the peak to  
 1317 illumination point correspondence ambiguities.

### 1318 A.3 Intensity Only Measurements

1319 One of the primary hypotheses of this work is that the temporal  
 1320 dimension of transient measurements contains information that  
 1321 can enable new advancements in 3D computer vision. Thus, for  
 1322 each of the tasks that we perform – estimation of depth, specular  
 1323 surfaces, and occluded geometry – we also conduct a baseline using  
 1324 an *intensity* image from a single-photon lidar. The intensity image  
 1325 is simply the sum of the transient along the temporal dimension.  
 1326 As expected, using the intensity image for training and inference  
 1327 on each task results in significantly worse performance than our  
 1328 method, which leverages the full information in the 3D transient.  
 1329 Using the intensity image results in 0.174 m mean absolute error for  
 1330 depth estimation, 0.703 IoU for specular segmentation, and 0.700 IoU  
 1331 for shadow mapping. While a significant drop in performance, inter-  
 1332 estingly, the intensity image still contains relevant cues for modest  
 1333 performance on each task, though we found the resulting shadow  
 1334 maps are not sufficiently accurate to perform 3D reconstruction.

### 1336 A.4 Two-Bounce Only Measurements

1337 We also study the impact of training on transients that only contain  
 1338 first and second bounce information to understand which signals  
 1339 the model has learned to exploit. We do this by re-rendering the  
 1340 transient dataset in MitsubaToF and setting *max\_depth* to three  
 1341 (whereas the main dataset contains all bounces). We re-train our  
 1342 models for each task, allowing us to understand the importance of  
 1343 three or more bounces of light based on the change in performance  
 1344 per task. Depth MAE increases from 0.0228 to 0.0255 m ( $\Delta 0.0027$  m),  
 1345 shadow mapping IoU increases from 0.954 to 0.964 ( $\Delta 0.01$ ), and spec-  
 1346 ular surface IoU drops from 0.865 to 0.767 ( $-\Delta 0.098$ ). Thus, while  
 1347 three bounce has a minimal impact on depth and shadows, it has  
 1348 a significant impact on specular surface segmentation. This result  
 1349 matches our intuition that three-bounce signals can contain infor-  
 1350 mation about specular surfaces. While two-bounce signals may  
 1351 indicate a specular surface based on the presence of an extra mea-  
 1352 sured peak at scene points that receive light reflected directly off a  
 1353 specular surface, three-bounce signals are empirically more help-  
 1354 ful. In particular, we posit that the model has not only learned to  
 1355 exploit diffuse-specular-diffuse light paths, as done in past work,  
 1356 but also diffuse-diffuse-specular light paths, which may be a fruitful  
 1357 direction to investigate in future work. This hypothesis is based  
 1358 on the diffuse-diffuse-specular signal that is visually evident when  
 1359 watching the light-in-flight transient videos.

### 1361 A.5 Noise and Timing Jitter

1362 Since our method is trained with simulated measurements, we ab-  
 1363 late its ability to work when trained on measurements with realistic  
 1364 pulse shapes, noise, and timing jitter. As done by Chen et al. [2020],  
 1365 we follow the protocol established in Hernandez et al. [2017] for  
 1366

1367  
 1368

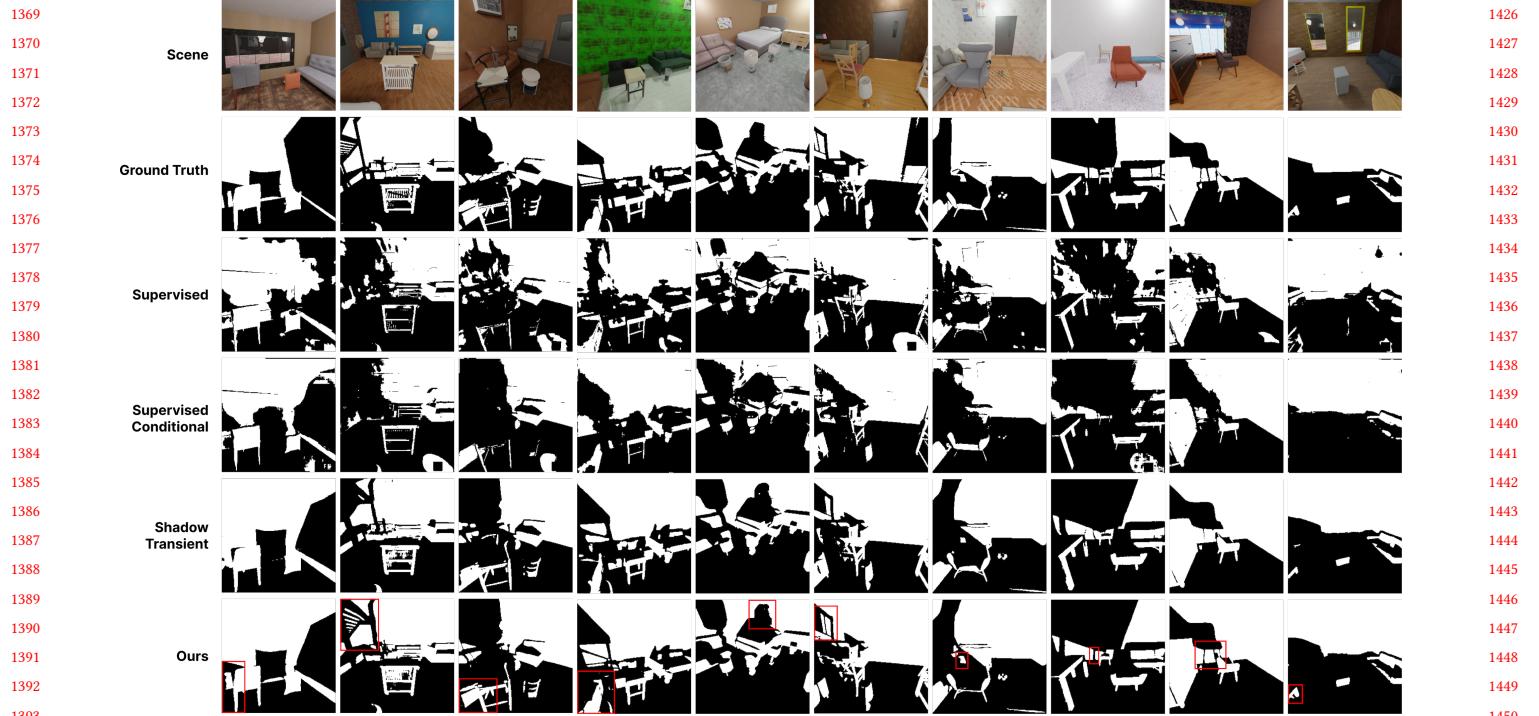


Fig. 12. **Shadow Mapping Approaches Ablation.** We compare different methods for predicting shadow masks from multiplexed lidar transients. All results are for measurements with 25 laser spots. We compare: supervised learning (transient  $\rightarrow$  25 shadows), supervised learning conditioned on laser spot index ( $(\text{transient}, \text{index}) \rightarrow$  shadow), shadow transients ( $\text{abs}(\text{2-Bounce ToF transient} - \text{measured transient}) \rightarrow$  shadow), and ours ( $(\text{2-Bounce ToF transient}, \text{measured transient}) \rightarrow$  shadow). Results indicate that the use of shadow transients – whether explicit or implicit (ours) significantly improves results. Providing both the predicted 2-bounce ToF transient and measured transient as input to the network, rather than explicitly computing the shadow transient, slightly improves detail and performance, as shown by the regions in the red boxes.

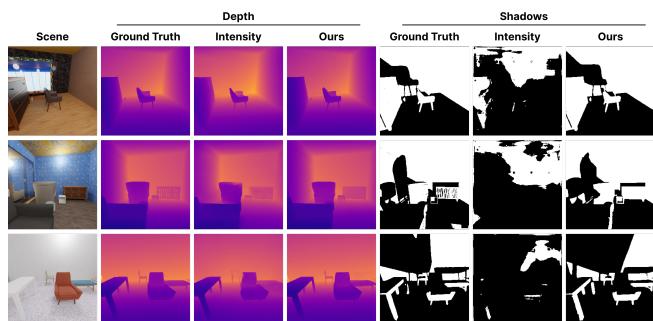


Fig. 13. **Intensity Only Ablation.** We compare depth and shadow estimation when using intensity images from the SPAD (obtained by summing along the temporal dimension) vs the full transient (ours). Using the full transient improves both depth and shadow results.

modeling realistic SPAD measurements. First, we convolve the rendered histograms with a pulse measured with a real-world sensor (MPD PDM Series). Next, we add Poisson noise and Gaussian timing jitter to the histograms. We sample the rate from a uniform distribution, leading to 2-bounce peak photon counts ranging from 10 to 400. We add 50 ps timing jitter (FWHM), which corresponds to



Fig. 14. **Two-Bounce Only Ablation.** We compare specular segmentation for models trained with transients rendered with only 1- and 2-Bounce peaks vs all peaks (ours). We randomly set pictures to be specular and find that only using 2-bounce information is not sufficient – thus, we posit this model also relies on 3-bounce information.

6.25 bins at 8 ps resolution. For this ablation, we use a small dataset containing geometric primitives to ease training time. The dataset consists of 10k training samples. Each scene also contains a mirror. We demonstrate both accurate depth estimation (which can then be

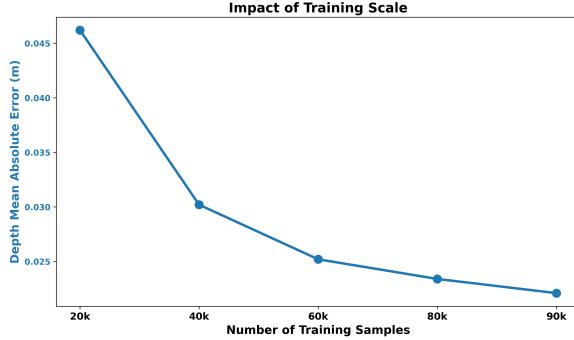


Fig. 15. **Ablation: Training Scale.** Given the scale of the proposed SB3D Dataset, we ablate the impact of training data scale on performance, focusing on depth accuracy. We find that as the amount of training data increases from 20k to 90k samples, the depth error consistency is reduced. Interestingly, the depth error reduces below the limit set by the temporal resolution of the sensor – meaning the model has learned precise correlations based on shape and appearance, rather than just relying on the timing information.

used to estimate two-bounce ToF) and shadow mapping using our method trained on this dataset. While our method is able to work in these conditions, we found that the 2D U-Net model was significantly less accurate than the “2.5D” U-Net alternative proposed in Appendix B. While the “2.5D” U-Net was not used in the main experiments due to higher training time, our use of a smaller and simpler dataset in this ablation, allowed us to use it. We posit that the 3D encoder enables the model to learn more robust temporal features, allowing it to generalize better under larger amounts of noise.

#### A.6 Impact of Training Scale

One of the contributions of this work is a large-scale simulated transient dataset. To ablate the impact of its scale on learning from multi-bounce signals, we vary the amount of training data used for our depth model and study the impact on performance. While past work in deep learning has extensively shown that data scale is correlated with performance, we use this ablation to confirm that intuition in the case of single-photon lidar data. We find that as we increase the amount of training data from 20k to 90k samples, depth estimation accuracy continues to improve, as shown in Fig. 15 of the main text.

## B Implementation Details & Architecture

**Pre-Processing.** Our method utilizes the raw multi-bounce lidar measurement as input, which, in our work had a shape  $256 \times 256 \times 637$  in our main experiments and  $256 \times 256 \times 375$  in our real-world experiments and noise ablation. We tried two methods for data normalization for the depth model: (1) reducing dynamic range by taking the log of each measurement followed by min-max normalization using the max intensity found over the entire dataset, and (2) min-max normalizing each histogram. While both were effective, we used (2) since it resulted in slightly better performance. For the shadow transient model, measured transients are min-max normalized and concatenated with the histogrammed predicted two-bounce ToF.

For real-world data, the measurements were instead  $256 \times 256 \times 375$  due to the differences in scene scale and temporal resolution.

**Architecture.** Although the focus of our work is not on architecture, we investigated the efficacy of different architectures for the proposed tasks, including 2D U-Net [Ronneberger et al. 2015], “2.5D” U-Net (3D encoder and 2D decoder with learned projections in each skip connection), SwinIR [Liang et al. 2021], NLOST [Li et al. 2023], and NLLOSFeatureEmbeddings [Chen et al. 2020]. We found that 2D U-Net and “2.5D” U-Net had the best performance – with 2D U-Net training faster since larger batches could be fit on GPU. Thus, we used a modified 2D U-Net for all results. To accommodate the large size of our input, we added an initial feature extraction convolution to project 637 bins (or 375 for real-world data) to 128 channels before proceeding to the six U-Net encoder and decoder blocks.

**Training.** In simulation-based experiments, we trained three models: a ToF demultiplexing model, a shadow demultiplexing model, and a specular surface segmentation decoder (using the frozen ToF demultiplexing features). All models were trained for 200 epochs. The shadow model was trained with ground-truth two-bounce ToF data for the first 100 epochs and then with the noisier predicted two-bounce ToF data for the last 100 epochs. We found this curriculum learning strategy to be most effective to maximize accuracy. While the depth model and shadow model were both trained to generalize over scenes, the neural reconstruction method [Klinghoffer et al. 2024] was trained per scene.

**Implementation.** Our models are implemented in PyTorch [Paszke et al. 2019] and each trained on 8 NVIDIA H100 GPUs for around two days due to the size of the dataset used. We use the AdamW optimizer [Loshchilov and Hutter 2019] with an initial learning rate of  $10^{-2}$  and weight decay of  $10^{-3}$ .

## C Training / Eval / Test Splits for SB3D Dataset

We train our models using the proposed dataset described in Sec. 5. We use 90% training split (87,688 samples), 3.9% validation split (3,744 samples), and 6.1% test split (6k samples). For 3D reconstruction, we train PlatoNerf per scene using the predicted two-bounce ToF and shadows, as described in Sec. 4. In all experiments we assume 25 illumination points in a grid pattern, unless stated otherwise (i.e., ablations on number of illumination points, noise ablation, and real-world experiments).

## D SB3D Dataset Rendering Details

Meshes of indoor scenes are created from objects in the Amazon Berkeley Objects dataset [Collins et al. 2022] following the method and pipeline for procedural generation of realistic indoor scenes proposed for the creation of the Aria Synthetic Environments dataset [Avetisyan et al. 2024]. Objects are assembled to mimic realistic indoor environments. These configurations were shown to be sufficiently realistic for real-world generalization of models trained on rendered RGB in past work [Avetisyan et al. 2024]. Besides single-photon lidar, all renders are created with Blender. Single-photon lidar transients are created with the physically-based MitsubaToF renderer [Pedireddla et al. 2019b], which uses bidirectional path tracing with ellipsoidal connections to increase sampling efficiency.

1597 Due to the computational complexity of rendering single-photon  
 1598 lidar and the scale of the proposed dataset, rendering was parallelized over 1,000 CPU machines over around one week. All data is  
 1599 rendered at a resolution of  $256 \times 256$  with a field of view of  $90^\circ$ . Multi-  
 1600 bounce lidar measurements are rendered with a temporal resolution  
 1601 of 128 picoseconds or  $\sim 0.0384$  meters. All bounces of light (1, 2, 3,  
 1602 and more) are rendered. To reduce rendering time, time gating is  
 1603 used when generating the transient data (all scenes have a minimum  
 1604 depth of no less than 0.5 m and a maximum depth of no more than  
 1605 4.5 m). To ensure all two-bounce paths are recorded, pathlengths  
 1606 between 1 and 25.46 meters are recorded, resulting in 637 bins per  
 1607 transient histogram. We set  $n_{bounces}$  to -1 in MitsubaToF, meaning  
 1608 all bounces of light are rendered. Thus, the dataset can be used  
 1609 in future work that explores additional bounces. For any specular  
 1610 surface, we use the "roughconductor" BSDF in MitsubaToF and set  
 1611 alpha to 0.01. As a result, the lidar transients have either diffuse  
 1612 or specular surfaces (but not a gradient). We acknowledge this is  
 1613 a limitation of the dataset, as, in practice, many real-world objects  
 1614 may exhibit material properties with partial diffuse and specular  
 1615 components, however, in our real-world proof-of-concept experiments,  
 1616 we find this assumption is sufficient in demonstrating the  
 1617 potential for real-world generalization. For accessibility, the dataset  
 1618 is compressed to  $\sim 5$  TB for release. We provide additional examples  
 1619 from the proposed SB3D dataset in Fig. 19.

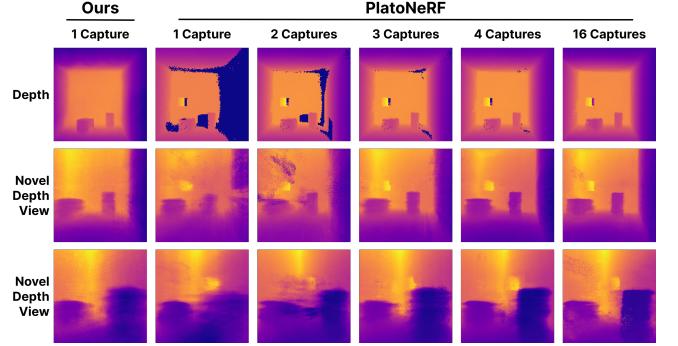
1620 Although not used in this work, the proposed dataset contains  
 1621 over 30 instance label categories that can be used in future work on  
 1622 instance segmentation, including everyday objects, such as desks,  
 1623 chairs, books, beds, pillows, and many more.

## 1626 E PlatoNeRF Background

1627 Our method leverages PlatoNeRF [Klinghoffer et al. 2024], a recent  
 1628 method for single-view 3D reconstruction from two-bounce  
 1629 transients. In contrast to our approach, PlatoNeRF assumes a laser is  
 1630 scanned over the scene sequentially, capturing separate transients  
 1631 for each laser spot. Each transient is preprocessed into two-bounce  
 1632 ToF and shadow masks, which are used to supervise the learned  
 1633 densities via volume rendering.

1634 PlatoNeRF is trained in two stages. First, depth from the lidar to  
 1635 the scene is learned by tracing primary rays with volume rendering.  
 1636 Since ground truth depth is not directly available, predicted depth  
 1637 is used to compute two-bounce ToF based on known illumination  
 1638 point and laser location (i.e. tracing the distance from the laser to  
 1639 the illumination point, from the illumination point to the predicted  
 1640 scene point location, and from the predicted scene point location to  
 1641 the known sensor location). In the second stage, secondary rays are  
 1642 also traced. Secondary rays originate at the end of primary rays and  
 1643 go to each illumination point. Intuitively, the secondary full ray path  
 1644 is only traveled if the measured point is *not* in shadow, else an object  
 1645 occludes the light from reaching the measured point. Thus, the  
 1646 secondary rays' transmittance values are supervised with the binary  
 1647 shadow masks extracted from the raw transient measurements.

1648 Since PlatoNeRF uses two-bounce ToF and shadow masks to learn  
 1649 3D scene geometry, the proposed pipeline naturally integrates with  
 1650 this approach for 3D reconstruction. Rather than computing these  
 1651 values from many non-multiplexed measurements, we instead use  
 1652



1654  
 1655  
 1656  
 1657  
 1658  
 1659  
 1660  
 1661  
 1662  
 1663  
 1664  
 1665  
 1666  
 1667  
 1668  
 1669  
 1670  
 1671  
 1672  
 1673  
 1674  
 1675  
 1676  
 1677  
 1678  
 1679  
 1680  
 1681  
 1682  
 1683  
 1684  
 1685  
 1686  
 1687  
 1688  
 1689  
 1690  
 1691  
 1692  
 1693  
 1694  
 1695  
 1696  
 1697  
 1698  
 1699  
 1700  
 1701  
 1702  
 1703  
 1704  
 1705  
 1706  
 1707  
 1708  
 1709  
 1710

Fig. 16. **PlatoNeRF Performance with More Captures.** While our method outperforms PlatoNeRF in the single capture setting, we compare to PlatoNeRF as more points are scanned and used in training. We find that, as expected, PlatoNeRF accuracy increases as more captures (with different illumination points) are added. Shoot-Bounce-3D remains competitive – higher performance is especially noticeable in specular regions, though occluded regions have more floaters.

the values predicted by our models from multiplexed measurements, enabling single-shot 3D reconstruction.

## F Real-World Experiments

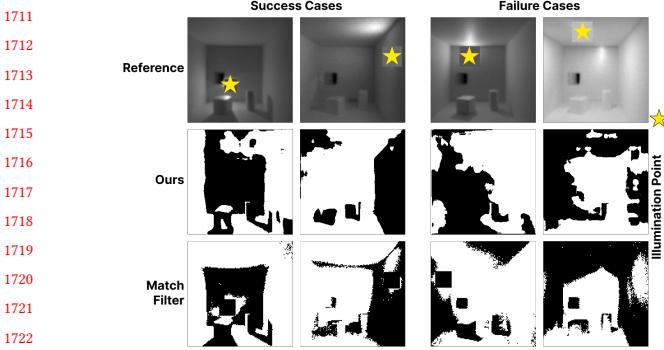
In this section, we elaborate on the details of our real-world dataset and proof-of-concept results.

### F.1 Model Training

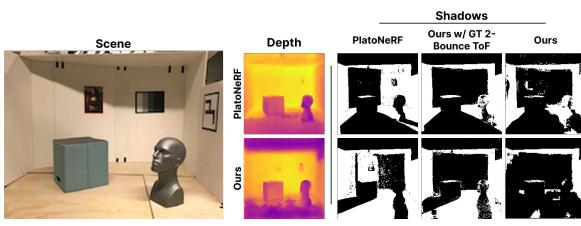
For real-world validation, we retrain our models on a simulated dataset of scenes with a randomly placed cube, cylinder, and mirror in a room of varying scale. The reason we retrain our models is to closer match the scale and type of geometry of the real-world scene that we captured (Sec. F.2). The simulated scene is illuminated at 16 points simultaneously – these points are in a grid pattern. When rendering, we randomly apply jitter to the camera origin and field of view for each scene. Each transient is rendered at 8 ps resolution and every four adjacent bins are summed to reduce the temporal resolution to 32 ps before training/inference. We choose 32 ps so that fine details can be resolved given the small scale of our scenes (the cylinder is one inch wide). Rendering with higher temporal resolution allows pulse shape and noise to be applied optionally before combining bins. The training dataset contained 10k samples, with an additional 2.5k for validation and 2.5k for test. We trained two sets of models – one with noiseless data and one with added pulse shapes, noise, and timing jitter (see Appendix A.5 for details).

### F.2 Real-World Dataset

We capture a real-world dataset with scene geometry and sensor intrinsics/extrinsics that lie in distribution with the training data described above. We construct a room from diffuse white poster board and randomly place a foam cube and cylinder inside it, along with a mirror on the back wall. We illuminate each laser spot one at a time with a pulsed laser (Picoquant LDH-D Series) with 640 nm wavelength and a two-axis scanning galvonometer (Thorlabs GVS412). For each laser spot, we then scan a single-pixel SPAD



**Fig. 17. Shadow Prediction Quality.** While our is able to produce shadows sufficient for 3D reconstruction, some predicted shadows have significant artifacts, as shown in the two examples on the right. Below our predicted shadows we show shadow quality when using a match filtr on the non-multiplexed measurement from the individual illumination point.



**Fig. 18. Limitations in Generalizability.** We test the generalizability of our models by testing them on an existing real-world dataset (from Bounce Flash Lidar [Henley et al. 2022]) with different scene scale, geometry, and spatial/temporal sensor resolution than seen in training. Our models are able to predict reasonable depth despite these differences. However, while some structure is maintained in predicted shadows, there are noticeable artifacts, especially in the region near the mannequin’s head. If we use ground truth 2-bounce ToF, along with raw transients, as input to our shadow model, instead of predicted 2-bounce ToF, shadow quality improves significantly. This improvement suggests that a limitation of our work is the propagation of errors from the depth estimation model to the shadow model.

(MPD PDM Series) over a 46° field of view using a second two-axis scanning galvonometer. This procedure results in sixteen 256 × 256 transients captured at 8 ps resolution. We sum adjacent bins in each transient to reduce the temporal resolution to 32 ps and add the sixteen transients together to create a multiplexed measurement. The light-in-flight video is shown in the supplementary webpage. Ground truth depth is captured from 1-bounce light by converting the setup shown in Fig. 7 to be confocal.

### F.3 Results

Results are shown in Fig. 7. While we found our noiseless and noised models were both able to estimate reasonable depth, recovered shadows varied in quality, with some being highly accurate and others containing more artifacts, as shown in Fig. 17. To find the best shadows, we performed a grid search over model (noiseless, noised), amount of noise to subtract, and maximum histogram intensity (for clipping). We also tried applying a low-pass filter to the data and

performing peak finding to reconstruct histograms with Diracs, but found that neither improved performance. We used the four best shadows, along with the 2-bounce ToF, predicted by our model to train PlatoNeRF for 3D reconstruction. In Fig. 7, we compare SB3D to PlatoNeRF trained when both are trained with only a single capture. Since PlatoNeRF is unable to handle multiplexed illumination, we instead train this PlatoNeRF model with a single illumination point. As shown in Fig. 16, as we increase the number of captures used to train PlatoNeRF, its performance improves. SB3D outperforms PlatoNeRF in the single capture setting and remains competitive with PlatoNeRF even when PlatoNeRF is trained with 16 captures. Training PlatoNeRF with 16 captures (by scanning a laser over different illumination points) serves as an upper bound on 3D reconstruction. With 16 captures, PlatoNeRF exhibits slightly fewer floaters/artifacts in occluded regions than SB3D, but SB3D exhibits better performance in areas with specular objects due to its use of a data prior.

*Limitations & Opportunities.* Our real-world results demonstrate feasibility that the ideas proposed in this work can extend to real-world settings. In this section, we investigate generalizability and limitations of our model.

We test our models’ ability to generalize to another real-world dataset, from BF Lidar [Henley et al. 2022]. This test is challenging because the scene in this dataset has different scale and geometry (e.g. a mannequin’s head) than the scenes in our training dataset. In addition, the multiplexed measurement from BF Lidar has different spatial and temporal resolution than our models were trained with. Specifically, our models from Sec. F.1 were trained with 256 × 256 spatial resolution and 32 ps temporal resolution, whereas the BF Lidar data is 200 × 200 spatial resolution with 128 ps temporal resolution. In addition, the BF Lidar scene is illuminated at 16 random points, rather than in a grid pattern. To account for this, we retrain the models described in Sec. F.1 with random illumination points for every training sample, testing whether our model can not only generalize to a real-world measurement with different geometry and resolution, but also generalize to random illumination patterns.

To test our models on the BF Lidar dataset, we zero-pad the measurements and rescale the detected two-bounce peaks bins based on the difference in scene scale between training and test. Results are shown in Fig. 18. Despite the significant domain gaps, our model is capable of predicting reasonable depth, albeit with artifacts. The predicted shadows contain accurate regions, but also regions with significant artifacts. To understand the cause of these artifacts, we tried using ground-truth 2-bounce ToF, rather than predictions, – along with raw lidar transients – as input to our shadow estimation model. This experiment resulted in significant improvements in shadow quality. This finding suggests that the shadow models are also able to generalize to different geometries and sensor resolutions if given accurate 2-bounce ToF, but errors in depth estimation propagate and can significantly impact the shadow model.

Future work may explore different types of noise to add to the two-bounce ToF during training to improve robustness or ways to unify the first two stages of our approach. Other improvements may come from incorporating real-world data into training and investigating other ways to mitigate the sim-to-real gap.

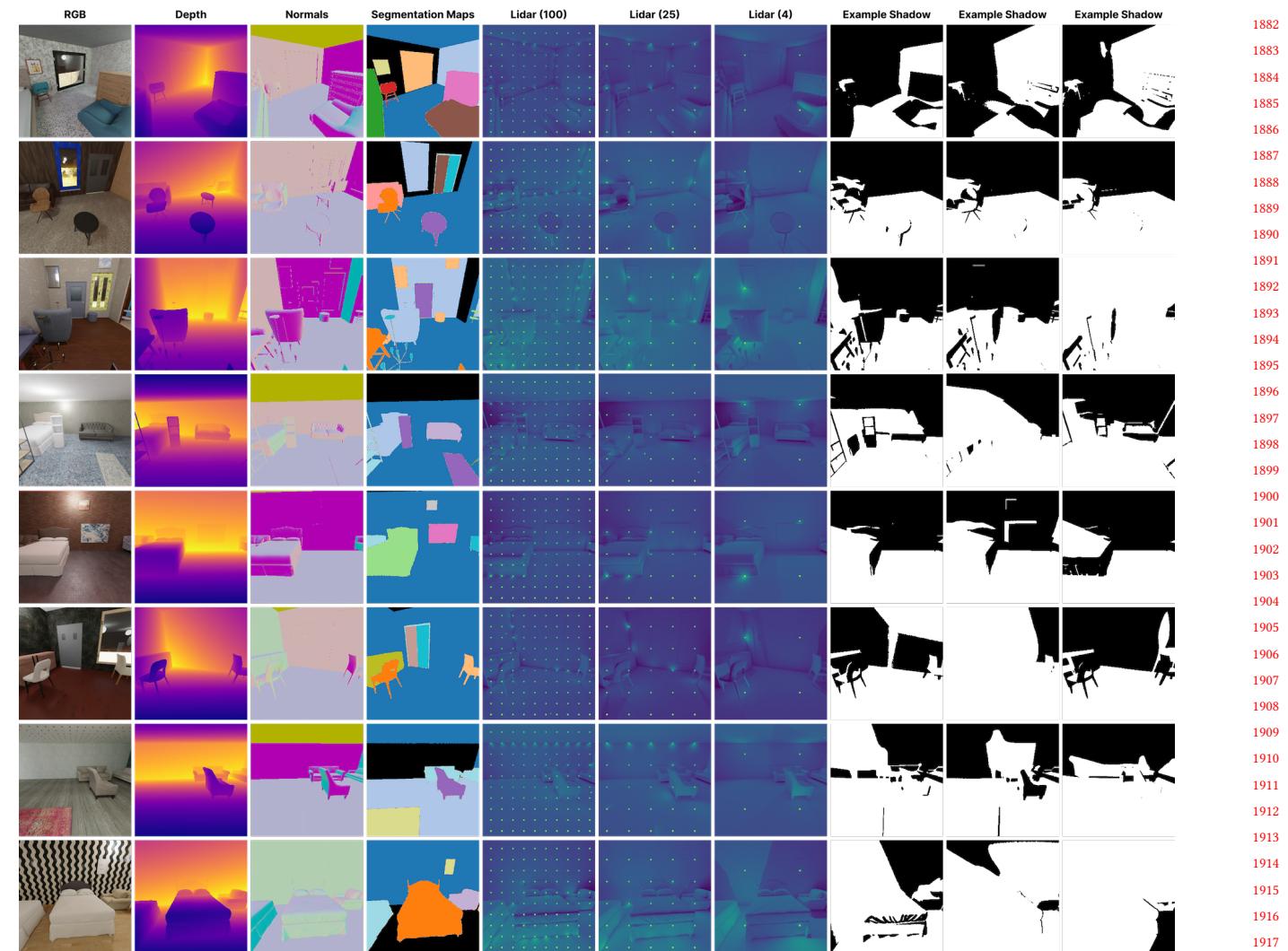


Fig. 19. **SB3D Dataset (Extended)**. We provide additional examples from our proposed dataset. In total, the dataset contains 97,432 examples, each rendered from a different scene.

1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938