Sparse Views, Near Light:
A Practical Paradigm for Uncalibrated Point-light Photometric Stereo

Supplementary Material

Abstract

In this supplementary material, we show further details about our framework. Specifically, we describe the network architecture with all its parameters and training specifications. Then we elaborate on the capturing process to retrieve the synthetic and real-world photometric images. After that, we show additional results on three viewpoints with a small camera baseline, as well as error maps of the reconstructions presented in the main paper. Furthermore, we show additional reconstruction results on both captured scans and the multiview diligent dataset, as well as some relighting results. We also analyze the effect of the ratio of point light intensity on the reconstruction quality, as well as the effect of the number of viewpoints and lights. Finally, we elaborate on the limitations of our approach.

1. Network Details

1.1. Architecture

As mentioned in the main paper, we use two multilayer perceptrons (MLPs). The first one describes the geometry via an SDF, d_θ, and the other one is used for the specular parameters of the material, α_γ. The MLP of d_θ consists of 6 layers of width 256, with a skip connection at the 4-th layer, while the MLP α_γ consist of 3 layers of width 256.
In order to compensate the spectral bias of MLPs [9], the input is encoded by positional encoding using 6 frequencies for both d_θ and α_γ. For the ablation *OurAlbedoNet*, a third MLP describing the BRDF's diffuse albedo, ρ_{γ_1}, is considered. It consists of 4 layers of width 512, and the input is encoded by positional encoding using 12 frequencies.

1.2. Parameters and Cost Function

Similarly to [1, 15, 18], we assume that the scene of interest lies within the unit sphere, which can be achieved by normalizing the camera positions appropriately. To approximate the Volume rendering integral (4) using (5), we use $m = 98$ samples which are also used to approximate (3), all with the sampling strategy of [16].
We set the objective's function trade-off parameters $\lambda_1 = \lambda_2 = 0.1$. Furthermore, the terms of the objective function (7) and (8) consist of a batch size of 800 (inside the silhouette) and 1000, respectively. For the mask term (9), we use the same batch as (7) and add 900 additional rays outside the silhouette whose rays still intersect with the unit sphere. Finally, we always normalize each objective function's summand with its corresponding batch size.

1.3. Training

Our networks are trained using the Adam optimizer [5] with a learning rate initialized with $5\text{e}-4$ and decayed exponentially during training to $5\text{e}-5$, except for the MLP α_γ whose learning rate is constantly equal to $1\text{e}-5$. The light positions ϕ are initialized with the camera position of their corresponding viewpoint, with a learning rate initialized with $1\text{e}-2$, and decayed exponentially with the same rate as the other networks. The remaining parameters are kept to Pytorch's default.
We train for 800 epochs, which lasts about 6 hours using a single NVIDIA Titan GTX GPU with 12GB memory and 6 viewpoints.

2. Data Acquisition

In this section, we describe how we generated the datasets used in this paper.

2.1. Synthetic Data

The synthetic datasets *dog1*, *dog2*, *girl1*, *girl2* were generated using Blender [3] and Matlab [8], where Blender [3] is used to render normal, depth and BRDF parameter maps for each viewpoint, and Matlab [8] is used to render images using equation (1) of the main paper. We used 20 point light illuminations for each viewpoint, with a ratio of 70% of point light intensity (thus 30% of ambient light), and we also added a zero-mean Gaussian noise with a standard deviation $\sigma = 0.02$.

2.2. Real World Data

In order to generate the real-world datasets *squirrel*, *bird*, *hawk*, *rooster*, *flamingo* and *pumpkin*, we used a Samsung Galaxy Note 8 and the application "CameraProfessional"[1] to generate RAW images as well as the smartphone's images in parallel. We use the RAW images for our algorithm, and we pre-processed those using Matlab [8] by following [12]. Since our approach assumes very precise camera parameters, and in order to facilitate calibration, we captured a higher amount of viewpoints and used COLMAP [11] to obtain both camera poses and intrinsics with the smartphone's images.

[1] https://play.google.com/store/apps/details?id=com.azheng.camera.professional

We move a hand-held LED[2] to obtain 20 images with different point light illumination per viewpoint.

3. Small camera baseline

As mentioned in the main paper, [7, 10, 17] lead to degenerate meshes when considering distant cameras, and are thus not suited to reconstruct full 3D objects from sparse viewpoints. For a more fair comparison, we focus here on a different scenario, where only a part of the object is reconstructed from three viewpoints with a very small camera baseline. Fig. 1 clearly indicates that our approach allows for much more accurate and complete 3D reconstruction than [7, 10, 17]. Note that all the meshes were obtained solely by using the official implementations.

4. Error maps

For a better appreciation of the quality of the full 3D reconstructions shown in the main paper, we show both the vertex-to-mesh distance and angular error maps in Fig. 2 and Fig. 3 respectively. We can see that our approach performs much better than the baseline at both the coarse and fine levels. Hence, it not only produces visually more pleasant reconstructions as can be seen in the main paper, but also with much higher fidelity.

5. Additional results

We can see in Fig. 4 the reconstruction results of our real-world scans that were not shown in the main paper. In order to further assess the quality of our framework on diverse materials, we performed an evaluation on the DiLiGenT-MV dataset [6]. Despite being captured with distant light sources, thereby satisfying the directional lighting assumption used in the baseline, our framework still achieves the best results both quantitatively and qualitatively as can be seen respectively in Tab. 1 and Fig. 5. Finally, we also show some relighting results in Fig. 7, together with the optimal diffuse albedo. This shows the validity of the estimated material parameters which can be successfully used for relighting, and indicates a proper disentanglement of the scene in terms of shape and material.

6. Effect of the ratio of point light

We further analyze the effect of the ratio of point light intensity on the quality of the result. This allows us to know how much ambient light can be handled by our approach while still providing accurate reconstructions. We remind that the total radiance can be decomposed into the sum of the point light radiance and the ambient light radiance, and we obtain the point light images by subtracting the input images with the ambient image. As discussed in section (3.2) of the main paper, one key issue with this strategy is that decreasing the amount of point light intensity yields point light images with a worse signal-to-noise ratio, which will inevitably affect the quality of the result. Consequently, we evaluate our approach on *dog2* using the same five viewpoints as in the main paper to obtain a full 3D reconstruction, with a point light intensity ratio ranging from 10% to 100% (dark room). We also consider two levels of noise, with standard deviations $\sigma \in \{0.02, 0.04\}$. As shown in Tab. 2 and Fig. 6, the quality of the result indeed improves as expected when increasing the amount of point light intensity. Moreover, for a given desired accuracy, a higher amount of point light is required for a noisier sensor, since this last yields the worst signal-to-noise ratio for the point light images. Nevertheless, even with a significant amount of noise, a reasonable result can be obtained starting from 40% of point light intensity, and a high accuracy with 70% and above. As mentioned in section (3.2) of the main paper, satisfying those requirements in practice is highly facilitated by the fact that near point lights are handled properly by our framework, in contrast to the majority of photometric stereo frameworks which require distant lights.

	↓MAE			↓RMSE×100		
	[2]	[14]	ours	[2]	[14]	ours
bear	19.1	8.8	**3.5**	2.2	1.0	**0.6**
buddha	39.6	13.9	**10.8**	2.5	0.7	**0.5**
pot2	29.2	9.2	**5.1**	10.2	0.7	**0.5**
reading	34.7	11.9	**7.1**	10.4	1.1	**0.9**
cow	25.3	8.9	**3.6**	6.1	0.9	**0.5**

Table 1. MAE and RMSE for the DiLiGenT-MV dataset [6].

7. Effect of the number of viewpoints and lights

Fig. 8 shows the MAE for both *dog2* and *girl2* using different numbers of viewpoints and light sources. Four viewpoints and five light sources allow to obtain a decent full 3D reconstruction, and six viewpoints and ten light sources are already enough for a high quality result.

8. Limitations

The optimal diffuse albedo allows to obtain great 3D reconstruction results in the most sparse scenarios. However, it is only defined for the viewpoints used for training, and is not multiview consistent, hindering novel view synthesis from arbitrary viewpoint. On the other hand, this issue is mitigated with our ablation *OurAlbedoNet* by using a neural diffuse albedo, at the cost of failing in some highly sparse scenarios. A straightforward solution would be to first use the optimal diffuse albedo strategy, then fix the geometry and specular parameters, and learn a neural diffuse albedo in a second stage. Successfully achieving multiview consistent

[2]We use white LUXEON Rebel LED: https://luxeonstar.com/product-category/led-modules/

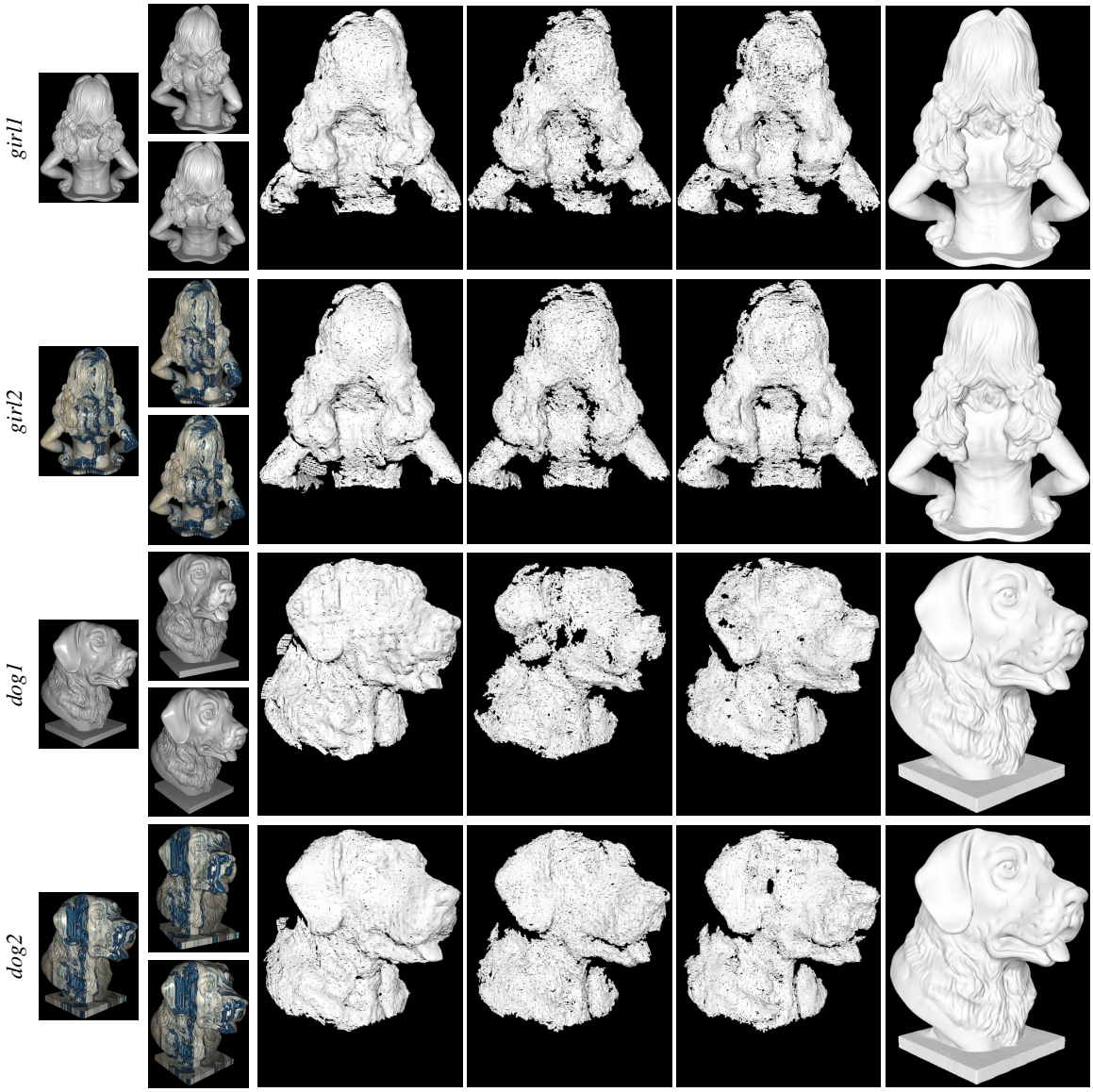


Figure 1. Results using three viewpoints with small camera baseline.

	↓RMSE×1000										↓MAE									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
$\sigma = 0.02$	10.3	5.6	4.3	3.9	3.3	3.4	3.2	3.2	2.9	**2.6**	12.7	7.8	6.3	5.7	5.2	5.0	4.7	4.8	4.6	**4.4**
$\sigma = 0.04$	16.1	10.3	6.5	5.2	4.5	3.9	3.6	3.6	3.5	**3.2**	16.7	12.8	9.4	7.8	6.9	6.2	5.8	5.5	5.4	**5.0**

Table 2. RMSE and MAE for different ratios of point light intensity, and two different levels of noise. RMSE is computed based on the vertex-to-mesh distance, and the MAE is computed using the angular error between the normals of a vertex and its closest point in the ground truth mesh.

diffuse albedo in the most sparse scenarios without relying on a second stage might increase the overall robustness, and is left as a future work. Moreover, we presume the availability of camera poses, acknowledging the challenge of pose estimation, particularly in the context of sparse viewpoints. A valuable extension of our work could be to address this assumption, *e.g.*, based on [4]. Finally, our BRDF choice is limited to opaque, non-metallic objects. Expanding our framework beyond those materials represents an intriguing avenue for future exploration.

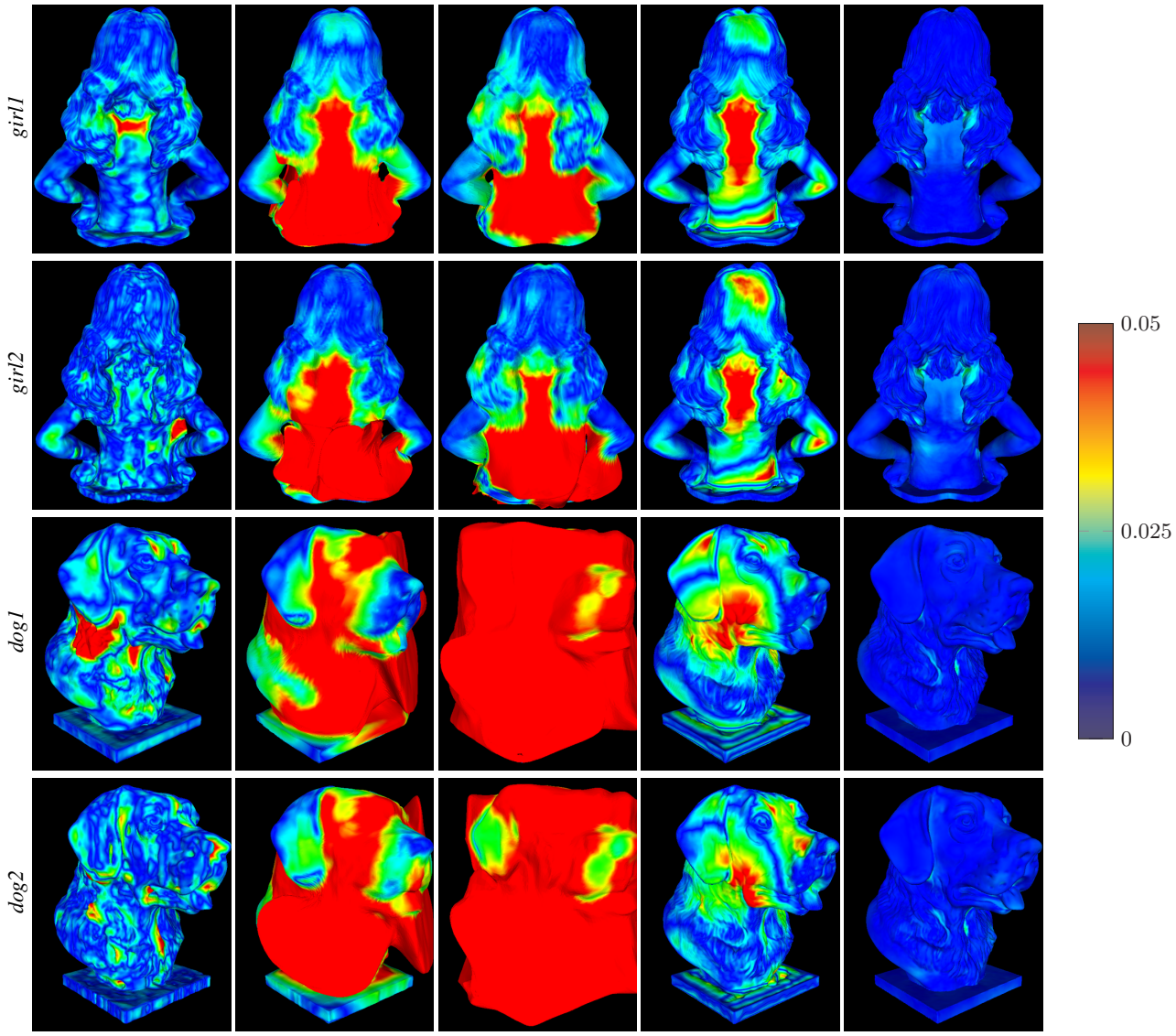


Figure 2. Vertex-to-mesh distance error maps. Errors are truncated for better visibility.

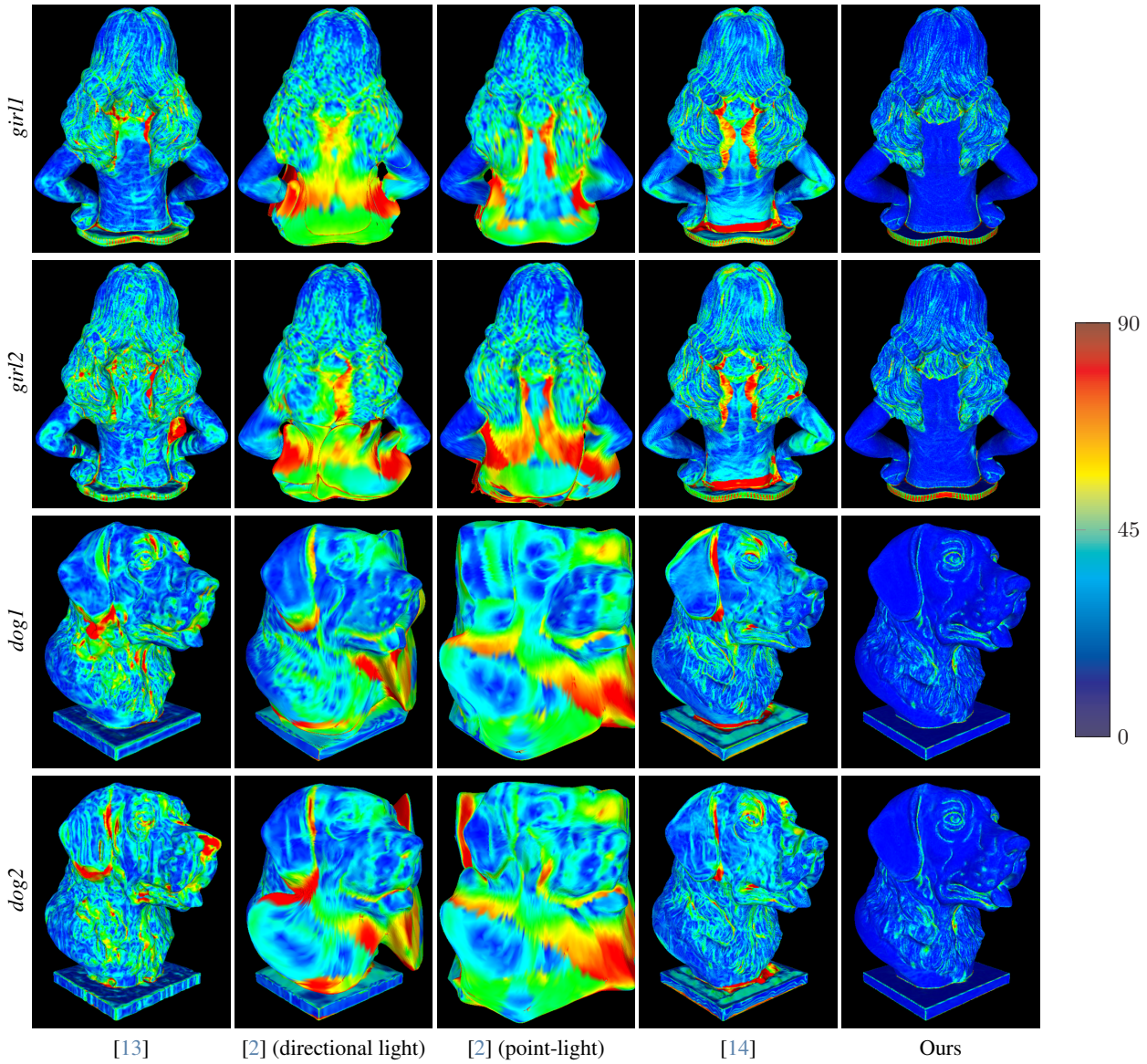
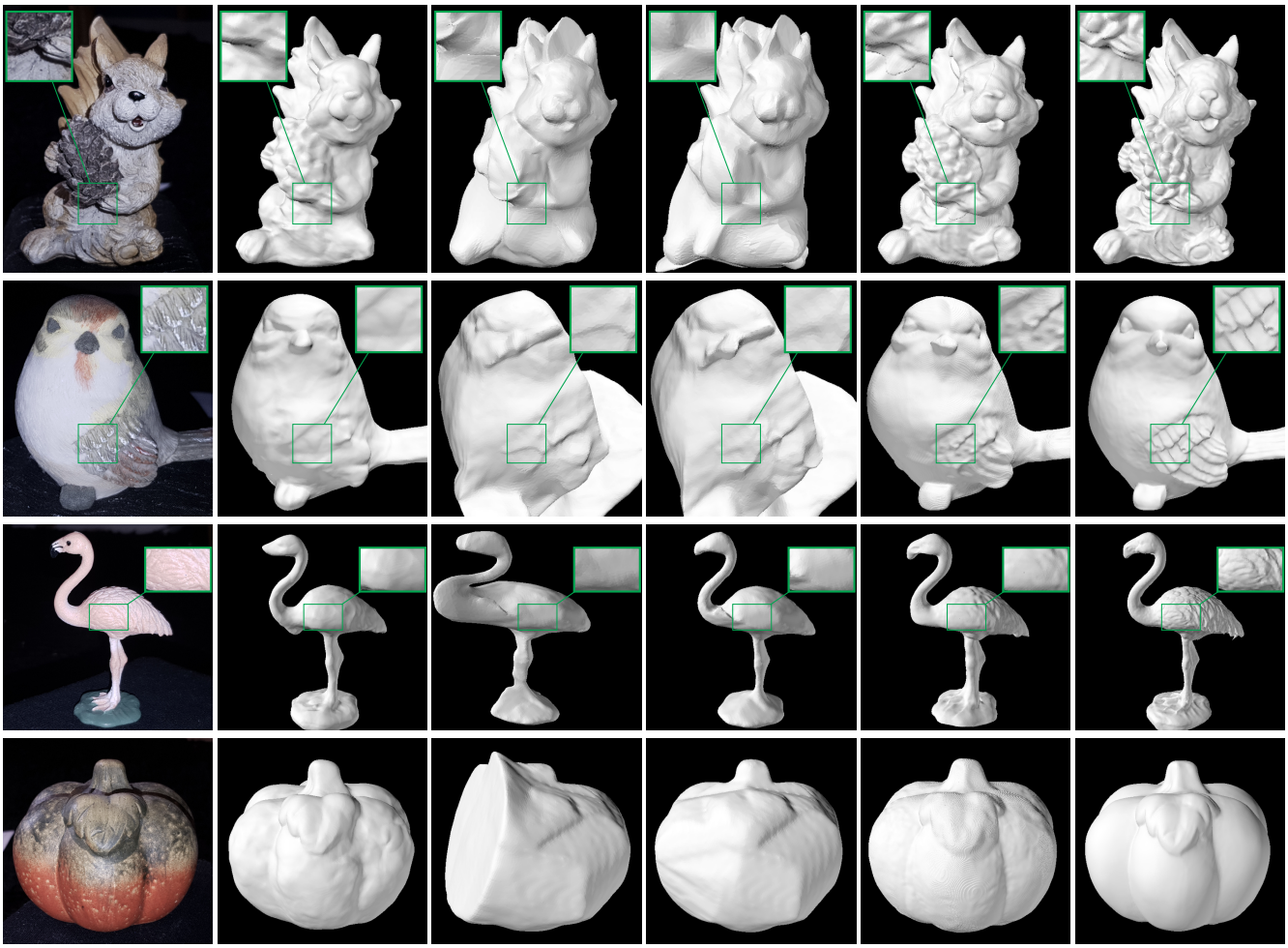
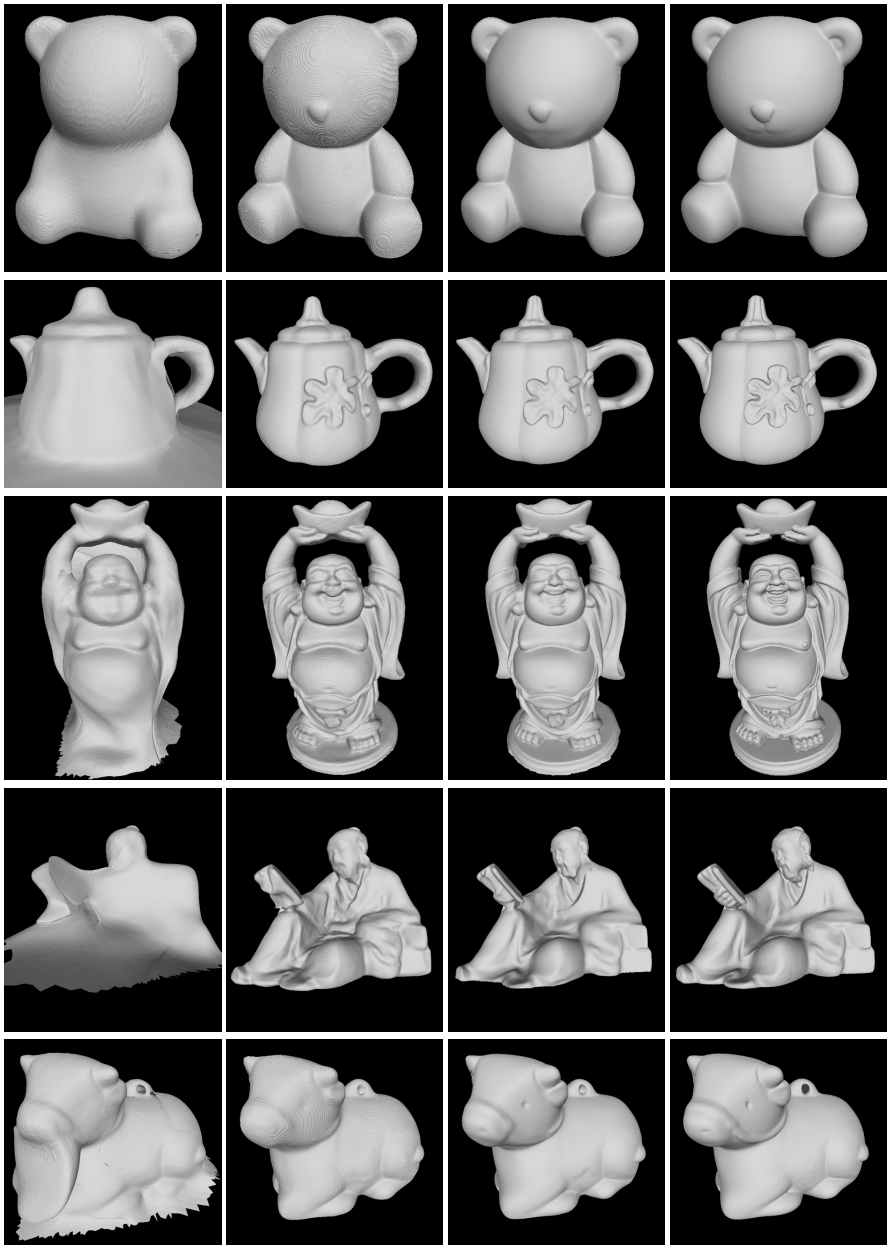


Figure 3. Angular error maps. Note that for *girl1* and *girl2*, a region of the plate at the bottom has significant errors for all approaches. This is due to the fact that in the ground truth mesh, vertices are only on the edges at that region, thus the angular error is not accurate there.



Input image [13] [2] (directional light) [2] (point-light) [14] Ours

Figure 4. Full 3D reconstruction of real objects from 6 viewpoints.



[2] [14] Ours Ground truth

Figure 5. Results on the DiLiGenT-MV dataset [6] using 6 viewpoints

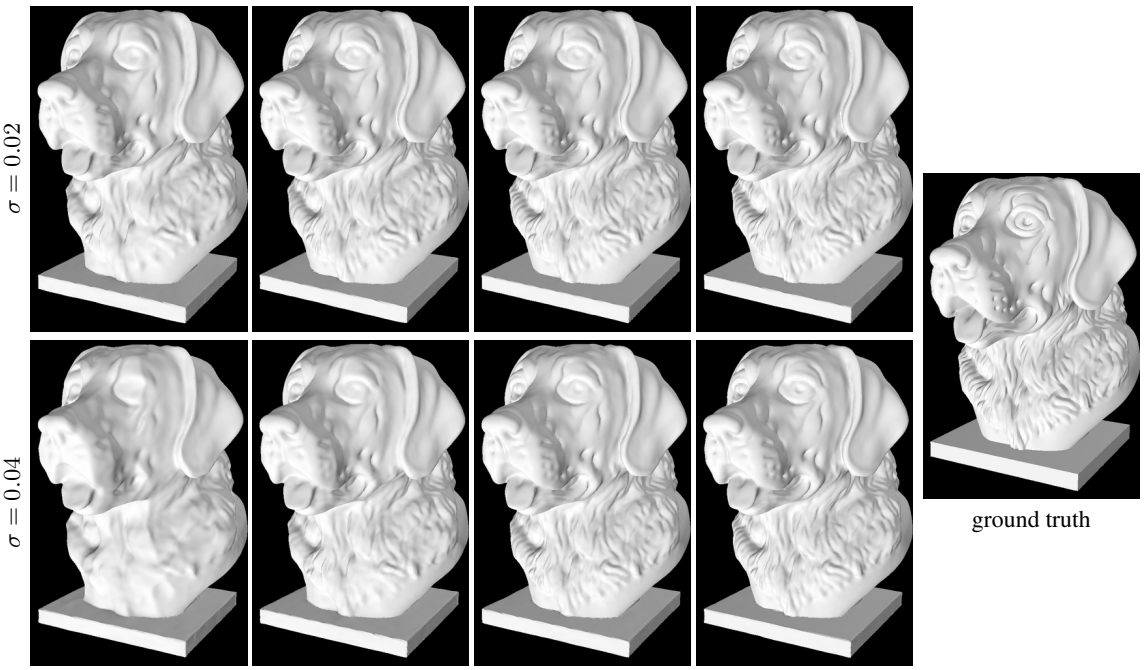


Figure 6. Results on *dog2* for different ratios of point light intensity. The first and second rows correspond to the results with Gaussian noise of standard deviation $\sigma = 0.02$ and 0.04 respectively.

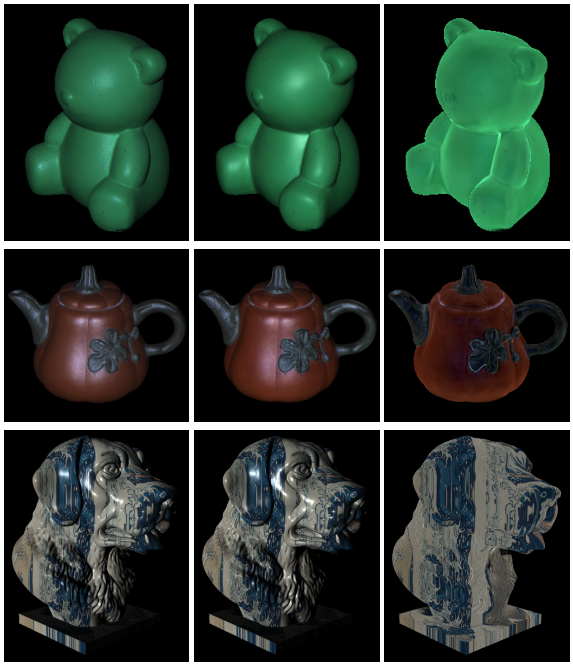


Figure 7. Relighting results.

	5L	10L	20L
4V	7.7	6.1	5.5
6V	6.4	5.0	4.6
8V	6.0	4.6	4.3
12V	5.1	4.2	3.9

(a) *dog2*

	5L	10L	20L
4V	13.7	12.4	12.4
6V	12.1	10.7	10.6
8V	11.3	10.3	10.1
12V	10.1	9.4	9.4

(b) *girl2*

Figure 8. MAE for different number of Viewpoints / Lights.

References

[1] Mohammed Brahimi, Bjoern Haefner, Tarun Yenamandra, Bastian Goldluecke, and Daniel Cremers. Supervol: Super-resolution shape and reflectance estimation in inverse volume rendering. *arXiv preprint arXiv:2212.04968*, 2022. 1

[2] Ziang Cheng, Hongdong Li, Richard Hartley, Yinqiang Zheng, and Imari Sato. Diffeomorphic neural surface parameterization for 3d and reflectance acquisition. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 2, 4, 5, 6, 7

[3] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 1

[4] Shi-Sheng Huang, Zi-Xin Zou, Yi-Chi Zhang, and Hua Huang. Sc-neus: Consistent neural surface reconstruction from sparse and noisy views. *arXiv preprint arXiv:2307.05892*, 2023. 3

[5] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[6] Min Li, Zhenglong Zhou, Zhe Wu, Boxin Shi, Changyu Diao, and Ping Tan. Multi-view photometric stereo: A robust solution and benchmark dataset for spatially varying isotropic materials. *IEEE Transactions on Image Processing*, 29:4159–4173, 2020. 2, 7

[7] Xiaoxiao Long, Cheng Lin, Peng Wang, Taku Komura, and Wenping Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, pages 210–227. Springer, 2022. 2, 3

[8] MATLAB. *version 9.8.0.1873465 (R2020a) Update 8*. The MathWorks Inc., Natick, Massachusetts, 2020. 1

[9] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1

[10] Yufan Ren, Tong Zhang, Marc Pollefeys, Sabine Süsstrunk, and Fangjinhua Wang. Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16685–16695, 2023. 2, 3

[11] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pages 501–518. Springer, 2016. 1

[12] Rob Sumner. Processing raw images in matlab. *Department of Electrical Engineering, University of California Sata Cruz*, 2014. 1

[13] Haoyu Wu, Alexandros Graikos, and Dimitris Samaras. S-volsdf: Sparse multi-view stereo regularization of neural implicit surfaces. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3556–3568, 2023. 4, 5, 6

[14] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K Wong. Ps-nerf: Neural inverse rendering for multi-view photometric stereo. In *European Conference on Computer Vision*, pages 266–284. Springer, 2022. 2, 4, 5, 6, 7

[15] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 1

[16] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. 1

[17] Ying-Cong Chen Yixun Liang, Hao He. Rethinking rendering in generalizable neural surface reconstruction: A learning-based solution. *arXiv*, 2023. 2, 3

[18] Kai Zhang, Fujun Luan, Zhengqi Li, and Noah Snavely. Iron: Inverse rendering by optimizing neural sdfs and materials from photometric images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 1