

Ref-GS: Directional Factorization for 2D Gaussian Splatting

Supplementary Material

This supplementary material provides additional information and experiment results pertaining to the main paper including detailed descriptions of the training process, and more visual results to complement the experiments reported in the main manuscript.

For more information regarding the method, we highly encourage readers to watch our video provided in the supplemental [webpage](#), where our method produces results with better specular reflection reconstruction.

A. Implementation Details

For training, we use the PyTorch [14] framework and train on a single Tesla V100 with 32GB of memory. Our code is build upon the 2D Gaussian Splatting (2DGS) [4] code-base. For real scenes, we propose using the same spherical domain strategy as 3DGS-DR [20] to train our model for a fair evaluation. This approach can reduce background interference during training. Background objects, captured from only limited viewpoints, exhibit similar behavior to reflective objects, which interferes with the fitting of our Sph-Mip.

A.1. Network

The goal of the shallow MLP f_{Θ} is to non-linearly map the directional feature $\mathbf{S} \in \mathbb{R}^{H \times W \times 16}$ produced by the Sph-Mip encoding and the high-dimensional intermediate tensor $\mathbf{K} \otimes \mathbf{S}$ has a shape of $H \times W \times 64$. Our MLP accepts an input having $16 + 64$ feature dimensions. The input is fed into a 2-layer MLP with 256 neurons per hidden layer in them followed by ReLU [1] activation functions. The output is fed into a output head predicts the view-dependent radiance with a exponential function output layer. Finally, we apply gamma tone mapping [2] $\gamma(\cdot)$ to convert the colors into the sRGB space before calculating the rendering loss:

$$\mathbf{I} = \gamma(\mathbf{I}_d + f_{\Theta}(\mathbf{S}, \mathbf{K} \otimes \mathbf{S})) \quad (1)$$

A.2. Optimization

The per-Gaussian position $\mu \in \mathbb{R}^3$, scale $s \in \mathbb{R}^2$ and covariance as rotation $q \in \mathbb{R}^4$, opacity $\alpha \in \mathbb{R}$, diffuse color $\mathbf{c}_d \in \mathbb{R}^3$, roughness $\rho \in [0, 1]$, feature $\mathbf{f} \in \mathbb{R}^4$ are optimized together with the network weights for the base MLP and the output head for view-dependent radiance. We use the Adam [7] optimizer with default parameters. Further, we follow the default splitting and pruning schedule proposed by the original 2DGS.

A.3. Losses

We have multiple loss terms in our training pipeline that are mainly adapted from 2DGS that we will briefly outline them and their weighting here. As in 2DGS, we use \mathcal{L}_1 loss and D-SSIM [18] loss for supervising RGB color, with $\lambda = 0.2$:

$$\mathcal{L}_{\text{rgb}} = (1 - \lambda)\mathcal{L}_1 + \lambda\mathcal{L}_{\text{D-SSIM}}. \quad (2)$$

Following 2DGS, depth distortion loss and normal consistency loss are adopted to refine the geometry property of the 2DGS representation of the scene.

$$\mathcal{L}_d = \sum_{i,j} \omega_i \omega_j |z_i - z_j| \quad \mathcal{L}_n = \sum_i \omega_i (1 - \mathbf{n}_i^\top \widehat{\mathbf{N}}) \quad (3)$$

Here, ω_i represents the blending weight of the i^{th} intersection. z_i denotes the depth of the intersection points. \mathbf{n}_i is the normal of the splat facing the camera. $\widehat{\mathbf{N}}$ is the normal estimated by the gradient of the depth map. The total loss is given as:

$$\mathcal{L} = \mathcal{L}_{\text{rgb}} + \lambda_d \mathcal{L}_d + \lambda_n \mathcal{L}_n \quad (4)$$

We empirically set $\lambda_d = 100$, $\lambda_n = 0.05$.

B. Limitations

While our approach demonstrates effective performance with a lightweight MLP for final color prediction, it results in slower rendering speeds compared to 2DGS and is challenging to integrate into standard CG rendering engines due to its reliance on a neural decoder. However, conversion techniques like textured mesh baking can facilitate integration and benefit from our reconstruction pipeline’s thin surface modeling and rendering capabilities.

C. Additional Results

In this section, we present additional visual results to demonstrate the capability of *Ref-GS* in reconstructing and rendering glossy surfaces, showcasing superior visual quality and accurate predicted normals for specular reflections across diverse scenes in the proposed dataset.

C.1. Shiny Blender Dataset

Tab. 1 provides the results on normal estimation for all scenes on Shiny Blender [17] dataset. For 3iGS [16], we use grad normals derived from the rendered depth map for evaluation.

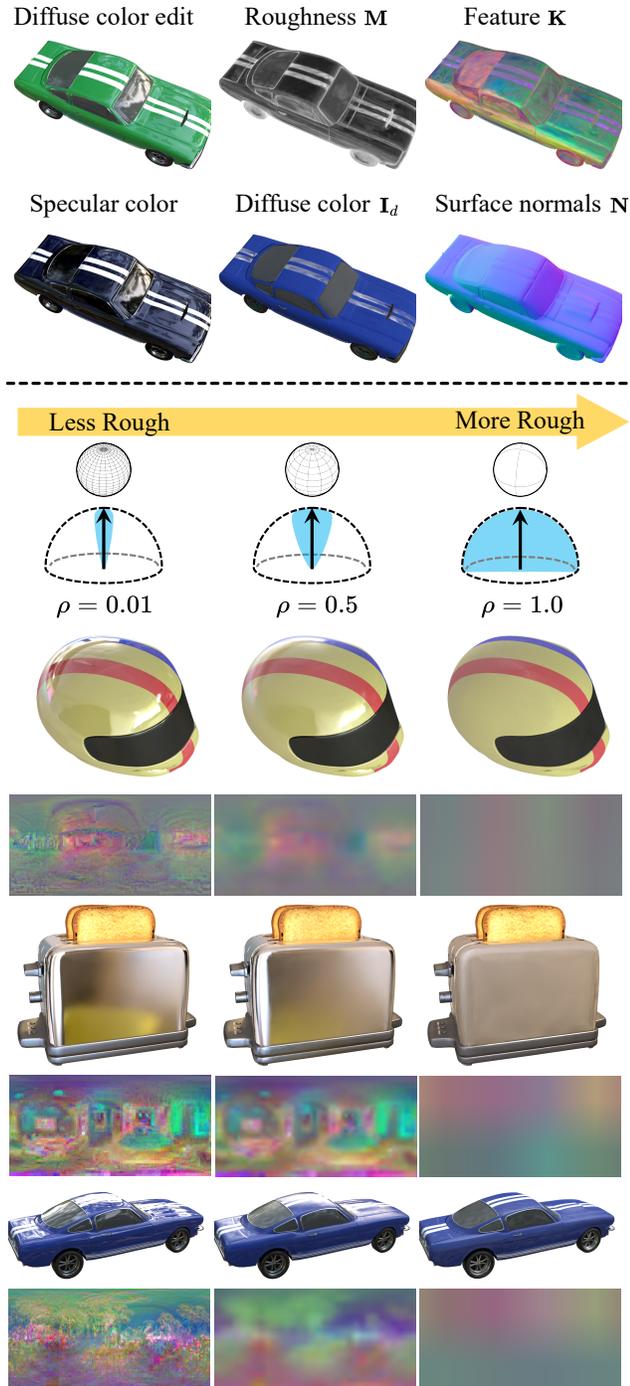


Figure 1. **Visualization of the Scene Decompositions and Material Editing.** Our model decomposes the appearance of synthetic scenes into interpretable components. *Ref-GS* effectively separates view-independent diffuse colors and view-dependent specular colors from multi-view training images. Furthermore, we can edit the diffuse color of the car without affecting the specular reflections on its glossy surface (top row). By modifying roughness ρ , we can obtain directional feature s at different levels can be obtained through Sph-Mip interpolation (bottom row).

	Shiny Blender						Avg.
	Car	Ball	Helmet	Teapot	Toaster	Coffee	
	MAE $^\circ$ \downarrow						
NVDiffRec [13]	11.78	32.67	21.19	5.55	16.04	15.05	17.05
Ref-NeRF [17]	14.93	1.55	29.48	9.23	42.87	12.24	18.38
ENVIDR [9]	7.10	0.74	1.66	2.47	6.45	9.23	4.61
GaussianShader [5]	14.05	7.03	9.33	7.17	13.08	14.93	10.93
GS-IR [10]	28.31	25.79	25.58	15.35	33.51	15.38	23.99
RelightGS [3]	26.02	22.44	19.63	9.21	28.17	13.39	19.81
3iGS [16]	11.79	31.78	16.72	2.61	21.12	8.80	15.47
3DGS-DR [20]	2.32	0.85	1.67	0.53	6.99	2.21	2.43
GS-ROR [22]	11.98	0.92	4.10	5.88	8.24	12.24	7.23
Ours	2.02	1.05	1.99	0.69	3.92	3.61	2.21

Table 1. Quantitative Mean Angular Error in degrees (MAE $^\circ$ \downarrow) of individual scenes on Shiny Blender [17] dataset. Red, Orange, and Yellow indicate the first, second, and third best performing methods for each scene.

C.2. Glossy Synthetic Dataset

We present the novel view synthesis results on the Glossy Synthetic [11] dataset. The quantitative evaluation in terms of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [18], and Learned Perceptual Image Patch Similarity (LPIPS) [21], is present in Tab. 2. Our approach outperforms the existing Gaussian-based methods [5, 16, 20, 22] on most scenes.

	Glossy Synthetic						
	Bell	Cat	Luyu	Potion	Tbell	Teapot	Avg.
	PSNR \uparrow						
Ref-NeRF [17]	30.02	29.76	25.42	30.11	26.91	22.77	27.50
NeRO [11]	—	—	—	—	—	—	—
ENVIDR [9]	30.88	31.04	28.03	32.11	28.64	26.77	29.58
3DGS [6]	25.11	31.36	26.97	30.16	23.88	21.51	26.50
GaussianShader [5]	28.07	31.81	27.18	30.09	24.48	23.58	27.54
3iGS [16]	25.60	30.93	27.17	29.50	23.94	21.17	26.39
3DGS-DR [20]	31.84	33.39	28.62	31.74	27.65	25.44	29.78
GS-ROR [22]	31.53	31.72	28.53	30.51	29.48	26.41	29.70
Ours	31.70	33.15	29.46	32.64	30.08	26.47	30.59
	SSIM \uparrow						
Ref-NeRF [17]	0.941	0.944	0.901	0.933	0.947	0.897	0.927
NeRO [11]	0.965	0.962	0.914	0.950	0.968	0.977	0.956
ENVIDR [9]	0.954	0.965	0.931	0.960	0.947	0.957	0.952
3DGS [6]	0.908	0.959	0.916	0.938	0.900	0.881	0.917
GaussianShader [5]	0.919	0.961	0.914	0.936	0.898	0.901	0.922
3iGS [16]	0.898	0.960	0.916	0.936	0.896	0.869	0.913
3DGS-DR [20]	0.964	0.976	0.938	0.957	0.948	0.939	0.954
GS-ROR [22]	0.969	0.967	0.938	0.950	0.965	0.947	0.956
Ours	0.965	0.973	0.946	0.957	0.956	0.944	0.957
	LPIPS \downarrow						
Ref-NeRF [17]	0.102	0.104	0.098	0.084	0.114	0.098	0.100
NeRO [11]	0.056	0.052	0.072	0.084	0.046	0.028	0.056
ENVIDR [9]	0.054	0.049	0.059	0.072	0.069	0.041	0.057
3DGS [6]	0.104	0.062	0.064	0.093	0.125	0.102	0.092
GaussianShader [5]	0.098	0.056	0.064	0.088	0.122	0.091	0.087
3iGS [16]	0.104	0.057	0.064	0.089	0.119	0.103	0.089
3DGS-DR [20]	0.044	0.039	0.052	0.073	0.070	0.062	0.057
GS-ROR [22]	—	—	—	—	—	—	—
Ours	0.049	0.041	0.046	0.076	0.073	0.064	0.058

Table 2. Quantitative results of individual scenes on Glossy Synthetic [11] dataset. Red, Orange, and Yellow indicate the first, second, and third best performing methods for each scene.

C.3. Glossy Real Dataset

We present the geometry reconstruction results on the Glossy Real [11] dataset to further validate the robustness and accuracy of our approach. We visualized the reconstruction results as shown in Fig. 2.

For a more comprehensive view of our method’s performance, please refer to the videos provided on the supplemental [webpage](#).



Figure 2. Images, ground-truth and reconstructed surfaces of the Glossy Real [11] dataset.

C.4. NeRF Synthetic Dataset

Quantitative results on the NeRF Synthetic [12] dataset are reported in Tab. 3. Our approach achieves numerically and visually comparable results with Gaussian-based methods [5, 16, 20, 22], demonstrating the effectiveness of our method in rendering general objects.

C.5. Additional Ablation Results

We provide more ablation results of on synthesized test in Tab. 4. To more clearly demonstrate the distinct advantages of the 2D Gaussian representation, we replaced 2DGS [4] with 3DGS [6], using the shortest axis as the plane normal while keeping the rest unchanged for comparison, as shown in the first two rows of Tab. 4. Furthermore, We have conducted ablation studies on the grid size N of Sph-Mip, as shown in Tab. 4. Notably, 3DGS-DR[20] improves the performance of GaussianShader[5] by introducing deferred shading with a simple shading model. “w/o $\mathbf{K} \otimes \mathbf{S}$ ” demonstrates that the Sph-Mip encoding can further enhance rendering quality. Additionally, the results of “w/o DS” demonstrate that our method outperforms the explicit BRDF of GaussianShader.

C.6. Additional Results on Real-World Captures

In this section, we extend the evaluation of our proposed method to include its performance on Rodriguez *et al.* [15] and Kopanas *et al.* [8] datasets. The qualitative comparison in Fig. 3 shows that *Ref-GS* extends well to real scenes,

	NeRF Synthetic								
	Chair	Drums	Lego	Mic	Materials	Ship	Hotdog	Ficus	Avg.
	PSNR↑								
NeRF [12]	33.00	25.01	32.54	32.91	29.62	28.65	36.18	30.13	31.01
Ref-NeRF [17]	33.98	25.43	35.10	33.65	27.10	29.24	37.04	28.74	31.29
VolSDF [19]	30.57	20.43	29.46	30.53	29.13	25.51	35.11	22.91	27.96
ENVIDR [9]	31.22	22.99	29.55	32.17	29.52	21.57	31.44	26.60	28.13
3DGS [6]	35.82	26.17	35.69	35.34	30.00	30.87	37.67	34.83	33.30
GaussianShader [5]	33.70	25.50	32.99	34.07	28.87	28.37	35.29	33.05	31.48
3iGS [16]	35.59	26.75	35.94	36.01	30.00	31.12	37.98	35.40	33.60
3DGS-DR [20]	35.60	25.31	32.94	31.97	29.65	29.07	35.58	28.03	31.02
Ours	34.66	26.33	36.26	35.76	30.99	29.67	37.39	34.52	33.20
	SSIM↑								
NeRF [12]	0.967	0.925	0.961	0.980	0.949	0.856	0.974	0.964	0.947
Ref-NeRF [17]	0.974	0.929	0.975	0.983	0.921	0.864	0.979	0.954	0.947
VolSDF [19]	0.949	0.893	0.951	0.969	0.954	0.842	0.972	0.929	0.932
ENVIDR [9]	0.976	0.930	0.961	0.984	0.968	0.855	0.963	0.987	0.953
3DGS [6]	0.987	0.954	0.983	0.991	0.960	0.907	0.985	0.987	0.969
GaussianShader [5]	0.980	0.945	0.972	0.989	0.951	0.881	0.980	0.982	0.960
3iGS [16]	0.987	0.955	0.983	0.992	0.961	0.908	0.986	0.989	0.970
3DGS-DR [20]	0.986	0.946	0.978	0.987	0.958	0.894	0.982	0.963	0.962
Ours	0.985	0.952	0.982	0.991	0.964	0.890	0.984	0.982	0.966
	LPIPS↓								
NeRF [12]	0.046	0.091	0.050	0.028	0.063	0.206	0.121	0.044	0.081
Ref-NeRF [17]	0.029	0.073	0.025	0.018	0.078	0.158	0.028	0.056	0.058
VolSDF [19]	0.056	0.119	0.054	0.191	0.048	0.191	0.043	0.068	0.096
ENVIDR [9]	0.031	0.080	0.054	0.021	0.045	0.228	0.072	0.010	0.068
3DGS [6]	0.012	0.037	0.016	0.006	0.034	0.106	0.020	0.012	0.030
GaussianShader [5]	0.019	0.045	0.026	0.009	0.046	0.148	0.029	0.017	0.042
3iGS [16]	0.012	0.036	0.015	0.005	0.034	0.102	0.019	0.010	0.029
3DGS-DR [20]	0.014	0.055	0.026	0.028	0.038	0.129	0.033	0.055	0.047
Ours	0.013	0.044	0.016	0.009	0.042	0.127	0.021	0.017	0.036

Table 3. Quantitative results of individual scenes on NeRF Synthetic [12] dataset. Red, Orange, and Yellow indicate the first, second, and third best performing methods for each scene.

	Chair	Drums	Lego	Mic	Materials	Ship	Hotdog	Ficus
Ours	34.66	26.33	36.26	35.76	30.99	29.67	37.39	34.52
w/ 3DGS	34.15	25.86	34.74	34.73	31.32	29.52	36.78	33.10
Sph-Mip $N=8$	34.67	26.34	35.83	35.23	30.91	29.26	37.19	34.11
Sph-Mip $N=7$	34.64	26.36	35.80	35.17	31.00	29.23	37.10	34.10
Sph-Mip $N=6$	35.65	26.17	35.74	35.04	30.39	29.16	37.13	34.10
w/o DS	33.75	25.85	33.99	35.16	29.25	28.89	36.11	32.15
w/o $\mathbf{K} \otimes \mathbf{S}$	34.08	25.71	35.19	34.21	29.77	29.10	36.62	32.47

Table 4. Per-scene PSNR comparison on NeRF Synthetic dataset. w/ 3DGS: Using 3DGS as the representation of our *Ref-GS* with the rest unchanged.

producing clearer specular reflections of the complex real-world environments compared to the existing Gaussian-based methods.

C.7. Scene Decompositions and Editing

Fig. 1 illustrates the rendering decomposition results of the scene. For reflective objects exhibiting strong specular effects, our approach can effectively decompose both the view-independent diffuse color and view-dependent specular color. Furthermore, the predicted material properties (*e.g.*, roughness ρ) and far-field lighting \mathcal{M} are also very reasonable. Additionally, we can plausibly modify the roughness of the scenes by adjusting the ρ values.

C.8. Supplementary Video Results

For a more comprehensive understanding of the performance of our approach, please refer to the supplementary videos provided. Additionally, we have created an interactive [webpage](#) to vividly showcase the capabilities of our approach.

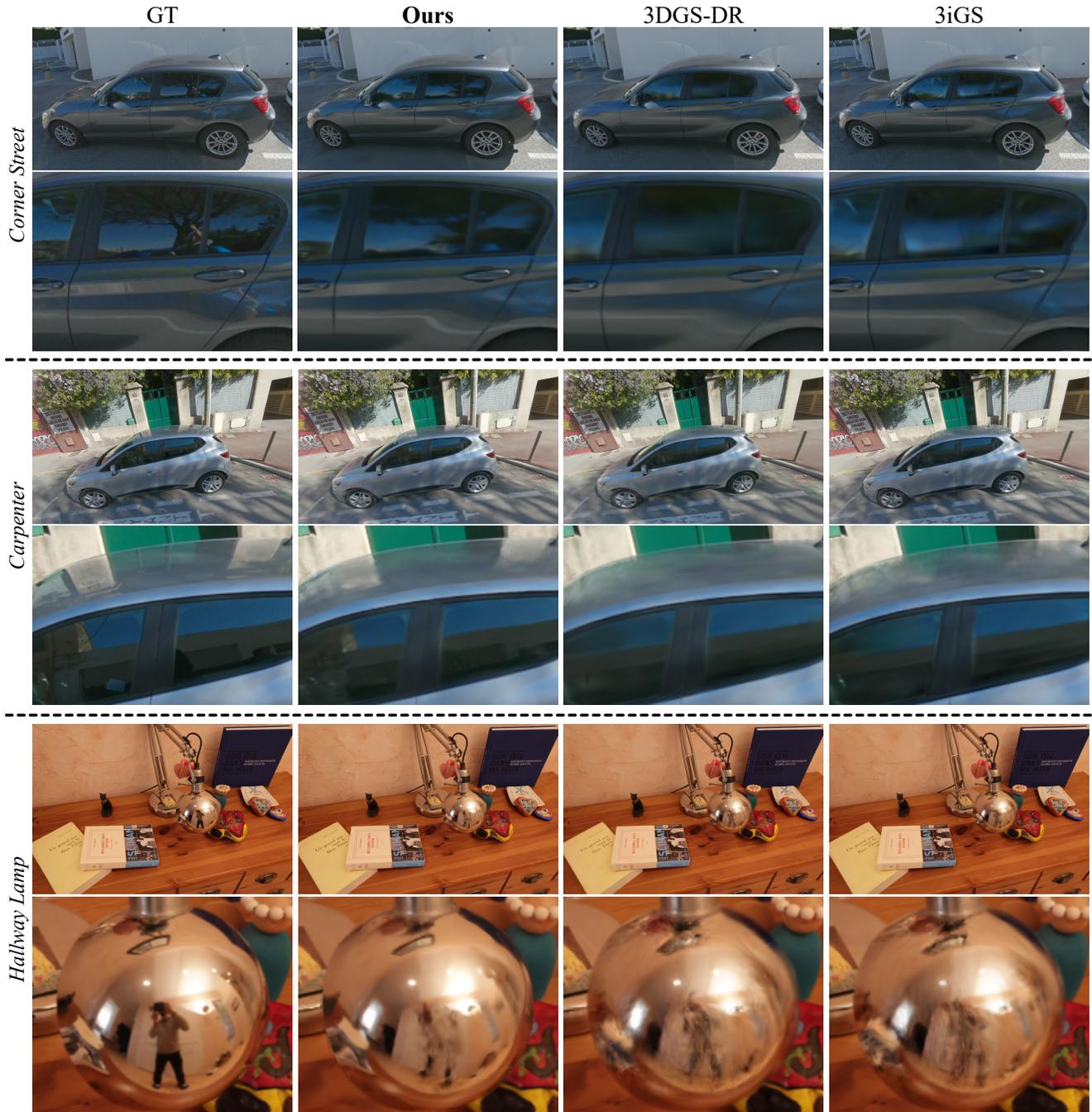


Figure 3. Additional results for intermediate component visualizations of our approach compared to 3DGS-DR [20] and 3iGS [16] on the Rodriguez *et al.* [15] and Kopanas *et al.* [8] datasets; zoom in to see the difference. (*Corner Street*, **1st row**) Our approach effectively simulates realistic reflections on the car body and windshield. (*Carpenter*, **2nd row**) Reflections of distant scenes on the car roof are rendered with impressive accuracy. (*Hallway Lamp*, **3rd row**) High-frequency details are well-preserved, enabling the realistic depiction of near-field content, including precise reflections.

References

[1] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018. 1

[2] Matthew Anderson, Ricardo Motta, Srinivasan Chandrasekar, and Michael Stokes. Proposal for a standard default color space for the internet—srgb. In *Color and imaging conference*, pages 238–245. Society of Imaging Science

- and Technology, 1996. 1
- [3] Jian Gao, Chun Gu, Youtian Lin, Zhihao Li, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao. Relightable 3d gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing. In *European Conference on Computer Vision*, pages 73–89. Springer, 2024. 2
- [4] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. 1, 3
- [5] Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. Gaussian-shader: 3d gaussian splatting with shading functions for reflective surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5322–5332, 2024. 2, 3
- [6] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. 2, 3
- [7] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1
- [8] Georgios Kopanas, Thomas Leimkühler, Gilles Rainer, Clément Jambon, and George Drettakis. Neural point catacaustics for novel-view synthesis of reflections. *ACM Transactions on Graphics*, 41(6):Article–201, 2022. 3, 4
- [9] Ruofan Liang, Huiting Chen, Chunlin Li, Fan Chen, Selvakumar Panneer, and Nandita Vijaykumar. Envidr: Implicit differentiable renderer with neural environment lighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 79–89, 2023. 2, 3
- [10] Zhihao Liang, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia. Gs-ir: 3d gaussian splatting for inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21644–21653, 2024. 2
- [11] 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. 2023. 2, 3
- [12] 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 *European Conference on Computer Vision*, pages 405–421. Springer, 2020. 3
- [13] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting Triangular 3D Models, Materials, and Lighting From Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8280–8290, 2022. 2
- [14] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. 2019. 1
- [15] Simon Rodriguez, Siddhant Prakash, Peter Hedman, and George Drettakis. Image-based rendering of cars using semantic labels and approximate reflection flow. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 3(1), 2020. 3, 4
- [16] Zhe Jun Tang and Tat-Jen Cham. 3igs: Factorised tensorial illumination for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 143–159. Springer, 2024. 1, 2, 3, 4
- [17] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022. 1, 2, 3
- [18] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 1, 2
- [19] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 3
- [20] Keyang Ye, Qiming Hou, and Kun Zhou. 3d gaussian splatting with deferred reflection. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–10, 2024. 1, 2, 3, 4
- [21] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 2
- [22] Zuo-Liang Zhu, Beibei Wang, and Jian Yang. Gs-ror: 3d gaussian splatting for reflective object relighting via sdf priors. *arXiv preprint arXiv:2406.18544*, 2024. 2, 3