PhyGaP: Physically-Grounded Gaussians with Polarization Cues

Supplementary Material

1. 3D Gaussian Splatting (3DGS) Preliminaries

The established 3DGS [7] represents a scene using 3D Gaussian Primitives $\mathcal{G}(\cdot)$, each characterized by a 3D mean position $\mathbf{p}$, a covariance matrix $\mathbf{\Sigma}$, an opacity o, and a set of spherical harmonics (SH) coefficients $\mathrm{SH}(\cdot)$ where view-dependent color $\mathbf{c}$ can be computed from viewpoint $\mathbf{r}$ as $\mathbf{c} = \mathrm{SH}(\mathbf{r})$.

To render an image, the 3D Gaussian is projected onto the camera space via:

$$p = \mathbf{JW}\mathbf{p}, \tag{1}$$

$$\Sigma_{\mathrm{2D}} = \mathbf{JW}\mathbf{\Sigma}\mathbf{W}^{\top}\mathbf{J}^{\top}, \tag{2}$$

where p and Σ_{2D} are the 2D mean and the 2D covariance matrix respectively, $\mathbf{W}$ is the world-to-camera transformation matrix, and $\mathbf{J}$ is the Jacobian matrix of the perspective projection. The influence of each Gaussian at 2D coordinates x is therefore defined as:

$$\mathcal{G}_{\mathrm{2D}}(x) = \exp\left[-\frac{1}{2}\left(x-p\right)^{\top}\Sigma_{\mathrm{2D}}^{-1}\left(x-p\right)\right]. \tag{3}$$

Next, volumetric α-blending is employed for each pixel x to blend $\alpha = o \cdot \mathcal{G}_{\mathrm{2D}}(x)$ in a front-to-back order to obtain the final color C, as follows:

$$C = \sum_{i=1}^{N} c_i \cdot T_i \cdot \alpha_i, \text{where } T_i = \prod_{j=1}^{i-1}(1-\alpha_j). \tag{4}$$

Other physical attributes associated with each Gaussian primitive in works like *Ref-Gaussian* [15], such as roughness and surface normal, can be α-blended into a feature map in a similar manner by replacing c_i in Eq. 4 with the target attribute a_i.

2. Detailed Formulation of PolarDR

2.1. Rendering Equation & Split-Sum Approx.

Based on Burley's reflectance model [2, 6] and Munkberg's derivation [11], the reflected radiance $L_o(\omega_o)$ in direction ω_o is given by:

$$L_{\omega_o} = \int_{\Omega} L_i(\omega_i) f_r(\omega_i, \omega_o)\langle\omega_i, \mathbf{n}\rangle \mathrm{d}\omega_i. \tag{5}$$

Notably, this equation integrates the product of incident light $L_i(\omega_i)$ from direction ω_i and the BRDF value $f_r(\omega_i, \omega_o)$. The integral is taken over the hemisphere Ω aligned with the surface normal $\mathbf{n}$. The BRDF term $f_r(\omega_i, \omega_o)$ can be further decomposed into diffuse and specular components, f_d and f_s, respectively.

Following *Ref-Gaussian*, we adopt the split-sum approximation [6] to cope with the intractable integral of *the specular component*:

$$\begin{aligned} L_s &= \int_{\Omega} f_s(\omega_i, \omega_o) L_s(\omega_i)\langle\omega_i, \mathbf{n}\rangle \mathrm{d}\omega_i \\ &= \int_{\Omega} \frac{DGF}{4(\omega_o, \mathbf{n})(\omega_i, \mathbf{n})} L_s(\omega_i)\langle\omega_i, \mathbf{n}\rangle \mathrm{d}\omega_i \\ &\approx \underbrace{\int_{\Omega} \frac{DGF}{4(\omega_o, \mathbf{n})(\omega_i, \mathbf{n})}\langle\omega_i, \mathbf{n}\rangle \mathrm{d}\omega_i}_{\text{Precomputed 2D Lookup}} \underbrace{\int_{\Omega} DL_s(\omega_i)\langle\omega_i, \mathbf{n}\rangle \mathrm{d}\omega_i}_{\text{Environment Mip-Map}}. \end{aligned} \tag{6}$$

Note that **the first term** is independent of the incident light. Thus, we precompute and store a 2D lookup table indexed by the roughness r and surface normal $\mathbf{n}$, which we can later query to realize $O(1)$ calculation of this term:

$$F_0 \cdot \tau_0 + \tau_1, \tag{7}$$

where $\tau_{\{0,1\}}$ are retrieved from the lookup table and F_0 refers to the Fresnel reflectance. For dieletric materials, F_0 is determined by the index of refraction (IoR) η:

$$F_0 = \frac{(1-\eta)^2}{(1+\eta)^2}. \tag{8}$$

The second term is represented with an environment cube mipmap $E(r, \omega)$ following Nvdiffrast [9] and Munkberg's implementation [11]. The base level possessing the minimum roughness r (0.08 in our case) represents the pre-integrated lighting with highest resolution, while each subsequent mip-level with increasing r is a filtered version of the previous one.

For *the diffuse component*, the BRDF is given by $f_d = \boldsymbol{\lambda}/\pi$ following the Lambertian model, where $\boldsymbol{\lambda}$ is the albedo. However, since we only account for the light emits from **subsurface scattering (SSS)**, we need to deduct the portion of light reflected (and therefore does not participate in SSS). Following Schlick's model, the Fresnel term F representing reflected energy is approximated by:

$$F = F_0 + (1-F_0)\cdot(1-\langle\omega_i', \mathbf{n}\rangle)^5. \tag{9}$$

The term ω_i' here is the direction of incident light reflected from the surface, different from the integrated ω_i. By excluding the diffuse pBRDF term f_d from the integration, the integral can be reformulated as:

$$L_d(\mathbf{n}) = (1-F)f_d \int_\Omega L_d(\omega_i)\langle\omega_i,\mathbf{n}\rangle \mathrm{d}\omega_i \approx (1-F)f_d \sum_{\langle\omega_i,\mathbf{n}\rangle>0} L_{\mathrm{env}}(\omega_i)\langle\omega_i,\mathbf{n}\rangle, \quad (10)$$

where $L_{\mathrm{env}}(\omega_i)$ is exactly $E(r_{\min},\omega_i)$, that is, the environment map with the smallest roughness (*i.e.*, highest resolution). Overall, the final pBRDF rendering equation can be expressed as:

$$L_{\omega_o} = (F_0\tau_0+\tau_1)E(r,\omega_i) + \frac{\lambda}{\pi}(1-F)\sum_{\langle\omega_i,\mathbf{n}\rangle>0} E(r_{\min},\omega_i)\langle\omega_i,\mathbf{n}\rangle.$$

2.2. Derivation of the pBRDF Model

The Fresnel transmission matrix $\mathbf{F}$ is written as:

$$\mathbf{F}^F = \begin{bmatrix} \frac{F^\perp+F^\parallel}{2} & \frac{F^\perp-F^\parallel}{2} & 0 & 0 \\ \frac{F^\perp-F^\parallel}{2} & \frac{F^\perp+F^\parallel}{2} & 0 & 0 \\ 0 & 0 & \sqrt{F^\perp F^\parallel}\cos\delta & \sqrt{F^\perp F^\parallel}\sin\delta \\ 0 & 0 & -\sqrt{F^\perp F^\parallel}\sin\delta & \sqrt{F^\perp F^\parallel}\cos\delta \end{bmatrix}, \quad (11)$$

where F can be either Fresnel transmission coefficients T or reflection coefficients R, and δ is the retardation phase shift between the perpendicular and parallel waves, either π or 0. T and R can be expressed as:

$$T^\perp = \left(\frac{2\eta_1\cos\theta_1}{\eta_1\cos\theta_1+\eta_2\cos\theta_2}\right)^2, \quad (12)$$

$$T^\parallel = \left(\frac{2\eta_1\cos\theta_1}{\eta_1\cos\theta_2+\eta_2\cos\theta_1}\right)^2, \quad (13)$$

$$R^\perp = \left(\frac{\eta_1\cos\theta_1-\eta_2\cos\theta_2}{\eta_1\cos\theta_1+\eta_2\cos\theta_2}\right)^2, \quad (14)$$

$$R_\parallel = \left(\frac{\eta_1\cos\theta_2-\eta_2\cos\theta_1}{\eta_1\cos\theta_2+\eta_2\cos\theta_1}\right)^2, \quad (15)$$

where $\perp$ and $\parallel$ indicate perpendicular and parallel waves in transmitted (T) and reflected (R) light; η_1, η_2 are the indices of refraction (IoR) of the medium before and after the interface, which are set to 1.0 and the object IoR η, respectively; θ_1 is the incident zenith angle, and $\cos\theta_2 = \sqrt{1-(\frac{\eta_1}{\eta_2})^2\sin^2\theta_1}$, from Snell's law.

An interesting fact is that according to the energy conservation law we always have the following:

$$\frac{\eta_2\cos\theta_2}{\eta_1\cos\theta_1}T^{\perp/\parallel} + R^{\perp/\parallel} = 1. \quad (16)$$

In order to transform Stokes vectors between the global coordinate system and the incident/exitant coordinate system, additional Mueller rotation matrices are applied before and after $\mathbf{F}$. A complete derivation of this can be found in [1]. The polarized shading models for diffuse and specular reflectance can be described as:

$$\mathbf{H}^d = \frac{\boldsymbol{\lambda}}{\pi}\langle\mathbf{n},\omega_i\rangle \cdot \begin{bmatrix} T_o^+T_i^+ & T_o^+T_i^-\beta_i & -T_o^+T_i^-\alpha_i & 0 \\ T_o^-T_i^+\beta_o & T_o^-T_i^-\beta_i\beta_o & -T_o^-T_i^-\alpha_i\beta_o & 0 \\ -T_o^-T_i^+\alpha_o & -T_o^-T_i^-\alpha_o\beta_i & T_o^-T_i^-\alpha_i\alpha_o & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (17)$$

$$\mathbf{H}^s = \frac{DG\langle\mathbf{n},\omega_i\rangle}{4\langle\mathbf{n},\omega_o\rangle\langle\mathbf{n},\omega_i\rangle} \cdot \begin{bmatrix} R^+ & R^-\gamma_i & -R^-\gamma_i & 0 \\ R^-\gamma_o & R^+\gamma_i\gamma_o+R^\times\chi_i\chi_o\cos\delta & -R^+\chi_i\gamma_o+R^\times\gamma_i\chi_o\cos\delta & \chi_oR^\times\sin\delta \\ -R^-\chi_o & -R^+\gamma_i\chi_o+R^\times\chi_i\gamma_o\cos\delta & R^+\chi_i\chi_o+R^\times\gamma_i\gamma_o\cos\delta & \gamma_oR^\times\sin\delta \\ 0 & -\chi_iR^\times\sin\delta & -\gamma_iR^\times\sin\delta & R^\times\cos\delta \end{bmatrix}. \quad (18)$$

Here $T_i^\pm = \frac{T^\perp\pm T^\parallel}{2}$ represent the Fresnel transmission coefficients into the surface, and $T_o^\pm$ represent the Fresnel transmission coefficients out of the surface after SSS. Similarly, $R^\pm = \frac{R^\perp\pm R^\parallel}{2}$ where R^+ is the portion of energy reflected on the surface, and $R^\times$ is irrelevant to our model. $\alpha_{i,o}$ and $\beta_{i,o}$ denote $\sin(2\phi_{i,o})$ and $\cos(2\phi_{i,o})$, where $\phi_{i,o}$ are the azimuth angles of incident/outgoing light. Following the assumption of PANDORA [3], the normal of microfacets $\mathbf{h}$ satisfies $\mathbf{h}=\mathbf{n}$, and $\chi_{i,o}$ and $\gamma_{i,o}$ are the same as $\alpha_{i,o}$ and $\beta_{i,o}$, respectively.

Specifically, for the diffuse $\mathbf{H}^d$, we suppose the scatter distance is small and no in-surface reflection take place, which means all energy participate in SSS will be emitted. Therefore, $T_o^+T_i^+$ can be equalized to the $1-F$ term in Eq. 10 since they both represent the ratio of light participating in SSS. Similarly, R^+ can be equalized to the ratio of reflected light, *i.e.* F in Section 2.1.

Suppose that incident light is unpolarized with Stokes vector $S_{in} = \begin{bmatrix} L_{\omega_i} & 0 & 0 & 0 \end{bmatrix}^\top$, the diffuse component of the outgoing Stokes vector can then be computed as:

$$\begin{aligned} S^d_{out}(\omega_i) &= \mathbf{H}^d \cdot \begin{bmatrix} L_{\omega_i} & 0 & 0 & 0 \end{bmatrix} \\ &= \frac{\boldsymbol{\lambda}}{\pi}\langle\mathbf{n},\omega_i\rangle \begin{bmatrix} T_o^+T_i^+ \\ T_o^-T_i^+\beta_o \\ -T_o^-T_i^+\alpha_o \\ 0 \end{bmatrix} \cdot L_{\omega_i} \\ &= (1-F)\cdot\frac{\boldsymbol{\lambda}}{\pi}\langle\mathbf{n},\omega_i\rangle \cdot \begin{bmatrix} 1 \\ (T_o^-/T_o^+)\beta_o \\ (-T_o^-/T_o^+)\alpha_o \\ 0 \end{bmatrix} \cdot L_{\omega_i}. \end{aligned} \quad (19)$$

Similarly, the specular component is:

$$S^s_{out}(\omega_i) = \mathbf{H}^s \cdot \begin{bmatrix} L_{\omega_i} & 0 & 0 & 0 \end{bmatrix}$$

$$= \frac{DGF}{4\langle \mathbf{n}, \omega_o \rangle \langle \mathbf{n}, \omega_i \rangle} \langle \mathbf{n}, \omega_i \rangle \begin{bmatrix} 1 \\ (R^-/R^+)\beta_o \\ (R^-/R^+)\alpha_o \\ 0 \end{bmatrix} \cdot L_{\omega_i}. \tag{20}$$

Please refer to the following papers for more details: Mitsuba3 [5], PANDORA [3], and pSVBRDF [1].

2.3. Training with Partially Polarization Cues

In cases where a polarization camera is unavailable, we may alternatively use regular RGB cameras overlaid with linear polarizers (LP) to acquire polarization information. The Muller Matrix of LP with orientation θ is written as:

$$\mathbf{M}_{\text{LP}}(\theta) = \frac{1}{2} \begin{bmatrix} 1 & \cos 2\theta & \sin 2\theta & 0 \\ \cos 2\theta & \cos^2 2\theta & \cos 2\theta \sin 2\theta & 0 \\ \sin 2\theta & \cos 2\theta \sin 2\theta & \sin^2 2\theta & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{21}$$

Therefore, we may formulate our Stokes loss between rendered Stokes S_{out} and captured light intensity I as:

$$\mathcal{L} = \|[\mathbf{M}_{\text{LP}}(\theta) \cdot S_{out}]_0 - I\|_1. \tag{22}$$

Note that θ is not necessarily known, and can be optimized alongside other parameters. A more detailed analysis can be found in Wu et al.'s work [14].

3. Training Details

3.1. Training Parameters

For most of our scenes with full polarization information, we set $\lambda_1 = 10$ and $\lambda_3 = 0.2$, except for *frog* and *dog* in the RMVP dataset, that we set $\lambda_1 = 1$ and $\lambda_3 = 0.3$. For all scenes, we set $\lambda_2 = 0.4$ and $\lambda_4 = 0.1$. For the index of refraction (IoR) η, we apply the activation function $\sigma(\mu) = \text{Sigmoid}(\mu) + 0.3$ to map it to $(1.3, 2.3)$, as 1.3 is the IoR of water and dielectric materials typically have IoR between 1.5 and 2.

3.2. Input Data

We convert all input images to linear color space to keep consistent with the rendering model. All results are converted to the sRGB color space for visualization. For GS-based methods, we hold out $1/8$ of all viewpoints as the test set for each scene.

3.3. Environment Map Details

We assume that environment maps encode light intensity at each pixel and therefore dwells in the linear color space. Consequently, we use the following activation function:

$$f(x) = \begin{cases} \text{Sigmoid}(x), & x \leq 0 \\ x + 0.5, & x > 0 \end{cases}. \tag{23}$$

All environment maps are converted to the sRGB color space for visualization.

3.4. GridMap Details

We update GridMap every 300 iterations to reflect changes in global illumination without causing too much computation overhead. The measured time overheads of GridMap are 24.1% and 26.8% for training and inference, respectively. We set $D = 64$ for cubemap construction.

In each one-step ray tracing that touches the object mesh, we retrieve the intersection point $\mathbf{p}$ and the surface normal $\mathbf{n}'$ of the mesh at $\mathbf{p}$. We perform the kNN algorithm to obtain an approximated albedo $\boldsymbol{\lambda}'$ at p, and use $\mathbf{n}'$ to query the global environment for global diffuse illumination $L_d(\mathbf{n}')$. Then, we calculate the local outgoing intensity of the object at p as $\mathbf{c} = \frac{\boldsymbol{\lambda}'}{\pi} L_d(\mathbf{n}')$ (essentially Eq. 10 without the Fresnel term). For rays that miss the object mesh, we directly copy the pixel values of the base environment map to construct the local cubemap.

We use a slightly larger bounding box, *i.e.*, $1.1\times$, for GridMap construction to avoid placing anchor cameras inside the object.

3.5. Data Acquisition Prototype Details

Our acquisition system features two RGB cameras, each overlaid with an linear polarizer (LP). The baseline between two cameras is set around 10 cm, and the LPs are rotated to approximately horizontal and vertical positions. During training, the rotation angles of the LPs are optimized to $170°\sim10°$ and $80°\sim100°$, respectively. We capture four scenes using this setup: *buddha-room*, *buddha-corridor*, *bull-corridor*, and *figurine-garden*, featuring objects of different materials (porcelain vs. plastic) and different environments (both indoor and outdoor). Each scene contains 50–80 viewpoints (*i.e.*, 100–160 images in total) that are uniformly sampled along a circular trajectory with radius $\sim$60 cm around the object. Noteworthy, careful readers may notice in the main text visualization that we set up an extra RGB camera (w/o LP) on our prototype rig, which is only to acquire data for baseline comparison matter.

The original image resolutions are $1,920 \times 1,200$, which we downsample to $1,152 \times 720$ during training. We apply the well-established COLMAP algorithm [12] to estimate camera poses, and use the langSAM and SAM [8] models to obtain object masks. Figure 1 visualizes reconstructed results of each scene using our PhyGaP framework.

4. Additional Results

Table 1 compares the SSIM↑ and LPIPS↓ scores of different methods on NVS, and Table 2 compares the MAE↓ scores on surface normal reconstruction.

Figure 7 visualizes additional reconstruction results of our PhyGaP method for different scenes. We observe that

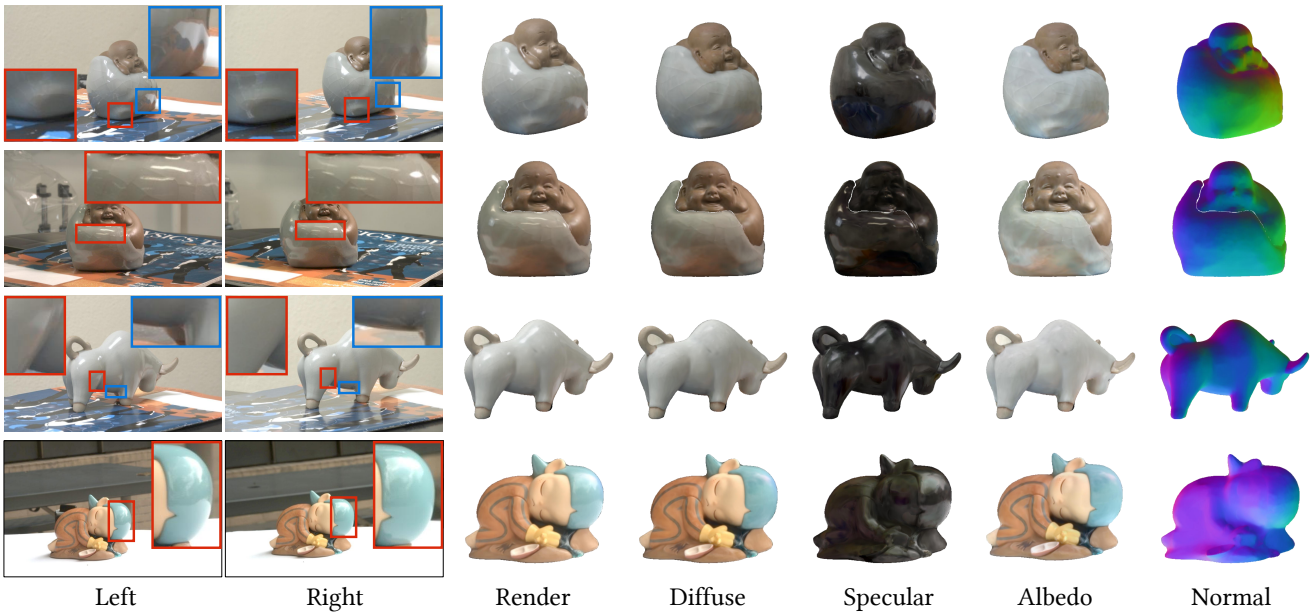

Figure 1. Real-word acquisition samples and reconstruction results. Note that reflection patterns captured by two cameras differ, due to the angular disparity between two LPs. From top to bottom: *buddha-room*, *buddha-corridor*, *bull-corridor*, and *figurine-garden*.

Table 1. SSIM↑ (*left*) and LPIPS↓ (*right*) of different methods on novel view synthesis. Best and second best results are indicated by **bold** and underline fonts respectively.

Methods	PANDORA		RMVP		SMVP			Mitsuba3	
	owl	vase	frog	dog	squirrel	snail	david	matpre.	teapot
R3DG	0.935 / 0.059	0.962 / 0.052	0.983 / 0.032	0.955 / 0.029	0.915 / 0.071	0.932 / 0.098	0.939 / 0.062	0.968 / 0.047	0.945 / 0.147
GS-IR	0.915 / 0.076	0.929 / 0.091	0.983 / 0.028	0.996 / 0.011	0.898 / 0.088	0.939 / 0.096	0.921 / 0.085	0.903 / 0.085	0.882 / 0.179
GIR	0.931 / 0.058	0.963 / 0.053	0.987 / 0.023	**0.999 / 0.003**	0.936 / 0.058	0.940 / 0.090	0.949 / 0.051	0.966 / 0.051	0.941 / 0.149
3DGS-DR	0.938 / 0.050	0.971 / 0.040	**0.991 / 0.018**	**0.999** / 0.006	0.933 / 0.057	0.947 / 0.075	0.952 / 0.043	0.968 / 0.046	0.947 / 0.135
Ref-Gaussian	0.924 / 0.064	0.972 / 0.039	0.988 / 0.028	**0.999** / 0.008	0.897 / 0.084	0.941 / 0.079	0.950 / 0.049	0.971 / 0.043	0.949 / 0.135
PolGS	0.941 / 0.064	0.960 / 0.064	0.965 / 0.043	0.991 / 0.022	0.926 / 0.071	0.921 / 0.126	0.966 / 0.038	-	-
PANDORA*	**0.960 / 0.042**	**0.984 / 0.038**	0.983 / 0.026	**0.997** / 0.008	**0.966 / 0.042**	**0.976** / 0.068	0.968 / 0.051	-	-
NeRSP *	-	-	-	-	0.958 / 0.049	0.975 / **0.059**	**0.979 / 0.030**	-	-
Ours	**0.960** / 0.043	0.975 / 0.048	0.988 / 0.024	**0.999** / 0.006	0.920 / 0.070	0.943 / 0.094	0.955 / 0.040	**0.977 / 0.042**	**0.957 / 0.131**

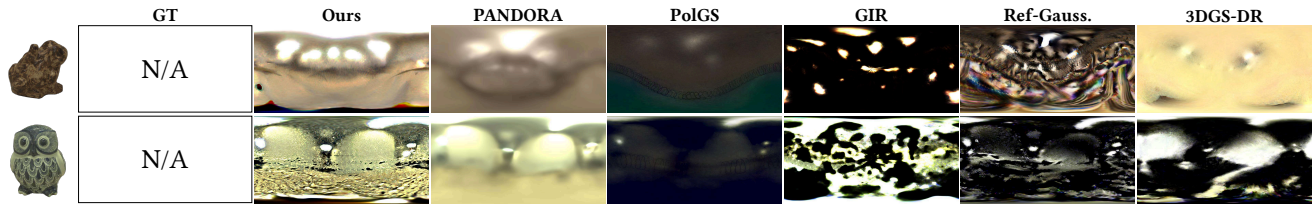

Figure 2. More qualitative comparison on estimated environment maps. Ground truths for these two scenes (*frog* and *owl*) are not available.

two RGB-LP cameras capture different reflection patterns, which enables our model to effectively decompose diffuse, specular and albedo, as well as to reconstruct smooth and realistic surface normals.

Additional comparisons on environment map estimation are available in Figure 2.

4.1. Evaluating Geometry Reconstruction

In addition to Cosine Distance and MAE that only validates surface normal consistency, we also measure Chamfer Distance (CD) to directly evaluate geometry reconstruction quality. For fair comparison, we firstly unproject the rasterized depth d of each pixel into the 3D space as $\mathbf{p} =$

Table 2. MAE↓ of different methods on surface normal reconstruction. Best and second best results are indicated by **bold** and underline fonts respectively.

Methods	RMVP		SMVP			Mitsuba3	
	frog	dog	squirrel	snail	david	matpre.	teapot
R3DG	17.46	24.86	19.90	13.22	21.12	18.22	11.62
GS-IR	21.35	27.31	21.13	20.86	21.62	22.45	19.26
GIR	15.98	29.74	13.91	14.04	24.18	13.61	7.90
3DGS-DR	19.89	24.96	12.01	13.03	24.06	20.64	9.21
Ref-Gaussian	14.01	27.43	17.12	8.35	21.23	9.01	4.97
PolGS	14.27	21.85	9.91	10.35	16.02	-	-
PANDORA*	14.28	**18.04**	**5.91**	18.45	15.51	-	-
NeRSP *	-	-	8.02	**5.52**	13.99	-	-
Ours	**13.58**	19.11	13.51	9.49	**13.72**	**8.32**	**4.03**

Table 3. Chamfer Distance on the SMVP dataset. Optimal results are indicated by **bold** font.

Scene	Chamfer Distance ↓ (mm)			
	Ours	PolGS	GIR	GS-IR
david	**6.535**	6.537	62.075	77.032
squirrel	9.527	**6.604**	13.118	28.148
snail	**8.627**	9.652	21.847	21.288

$d \cdot \mathbf{v} + \mathbf{p}_{\text{cam}}$, where $\mathbf{v}$ is the camera viewing ray and $\mathbf{p}_{\text{cam}}$ is the camera center, and then use Poisson surface reconstruction with depth 8 and pruning threshold 5×10^{-3} to generate meshes.

We compare our method against three GS-based baselines that produce relatively high-quality meshes with this process, namely: PolGS [4], GS-IR [10], and GIR [13]. Unidirectional (from predicted mesh to ground truth mesh, within the bounding box of the ground truth mesh) L1 distances are calculated.

As shown in Table 3 and Figure 3, we achieve comparable results as the previous state-of-the-art, *i.e.*, PolGS, and beats GIR and GS-IR by a large margin.

4.2. GridMap vs. SH-based Indirect Light

In Figure 4, we compare between SH-based indirect light modeling (as utilized in Ref-Gaussian [15]) and our GridMap solution. We observe that the former leads to inconsistent artifacts during relighting, while our GridMap produces realistic transition from directly illuminated regions to those influenced by indirect light.

4.3. Ablation on λ Choices

We conduct ablation study on different values of λs to show that the reported values in the main paper are sufficiently close to optimality. See Figure 5 for details.

*Training data for PANDORA and NeRSP models contain test viewpoints, and thereby their scores are only weakly comparable to others.

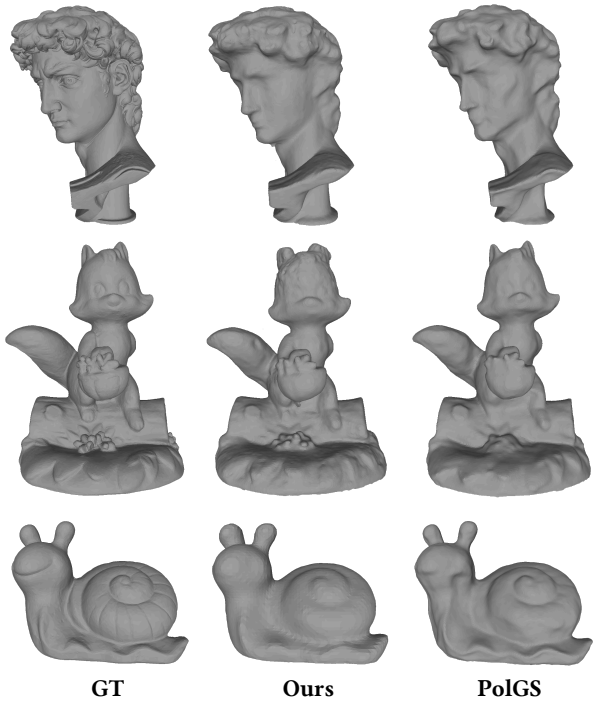

Figure 3. Meshes reconstructed by PhyGaP and PolGS. Results for GIR and GS-IR are not presented since their meshes exhibit significantly worse quality due to floating Gaussian points.

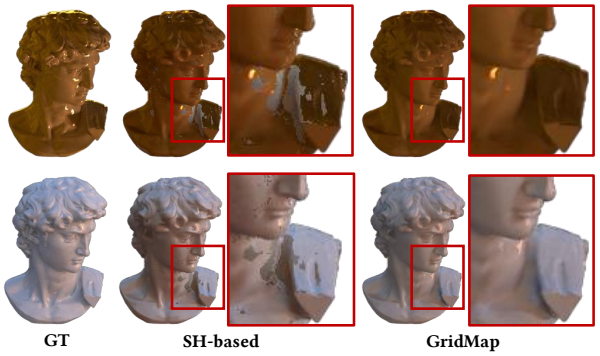

Figure 4. Relighting results using SH-based indrect light modeling and our GridMap.

4.4. Failure Case and Comparison between Anchor Placement Strategies

Our current strategy of placing anchor cameras on the bounding box of each object may fail under self-occlusion or objects with extreme shapes (*e.g.* very thin or deeply concave). We observe this to be the main cause of failure during relighting in our experiments, and we have visually demonstrate it in Figure 6.

Alternatively, we may choose to place anchors close to the object surface. However, this strategy also faces two issues: (a) Adaptively calculating anchor locations may in-

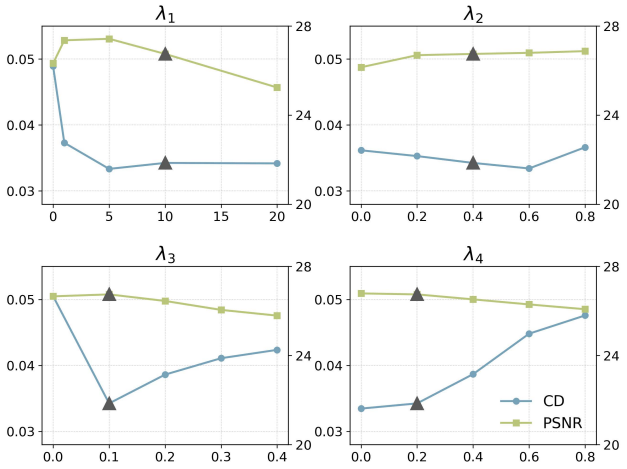

Figure 5. Cosine distance (CD↓, left axis) and PSNR↑ (right axis) across λ choices. The reported values are indicated by triangles.

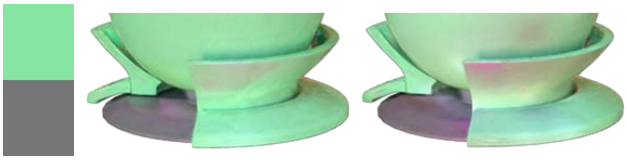

Figure 6. *Left* ground-truth colors for object body and stand. *Mid* albedo reconstructed using GridMap. Object concavity and self-occlusion result in reddish artifacts. *Right* albedo reconstructed using near-surface anchor placement. More severe color shifts are observed due to anchors being placed inside the object.

troduce additional computation overhead during training. (b) Meshes constructed from TSDF do not fully align with Gaussian geometries, which can lead to anchors being sampled inside objects under this strategy and cause even larger errors. We compare two anchor placement strategies in Figure 6 to validate the latter claim.

5. Supplementary Video

Refer also to the supplementary video `relight.mp4` for more relighting results.

References

[1] Seung-Hwan Baek, Daniel S Jeon, Xin Tong, and Min H Kim. Simultaneous acquisition of polarimetric SVBRDF and normals. *ACM Trans. Graph.*, 37(6):268, 2018. 2, 3

[2] By Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. 2012(2012):1–7, 2012. 1

[3] Akshat Dave, Yongyi Zhao, and Ashok Veeraraghavan. Pandora: Polarization-aided neural decomposition of radiance. In *European conference on computer vision*, pages 538–556. Springer, 2022. 2, 3

[4] Yufei Han, Bowen Tie, Heng Guo, Youwei Lyu, Si Li, Boxin Shi, Yunpeng Jia, and Zhanyu Ma. Polgs: Polarimetric gaussian splatting for fast reflective surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28073–28082, 2025. 5

[5] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org. 3

[6] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 4(3):1, 2013. 1

[7] 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. 1

[8] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 3

[9] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020. 1

[10] Zhihao Liang, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia. GS-IR: 3D Gaussian Splatting for Inverse Rendering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21644–21653, 2024. 5

[11] 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*, pages 8280–8290, 2022. 1

[12] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 3

[13] Yahao Shi, Yanmin Wu, Chenming Wu, Xing Liu, Chen Zhao, Haocheng Feng, Jian Zhang, Bin Zhou, Errui Ding, and Jingdong Wang. Gir: 3d gaussian inverse rendering for relightable scene factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 5

[14] Bojian Wu, Yifan Peng, Ruizhen Hu, and Xiaowei Zhou. Glossy object reconstruction with cost-effective polarized acquisition. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 422–431, 2025. 3

[15] Yuxuan Yao, Zixuan Zeng, Chun Gu, Xiatian Zhu, and Li Zhang. Reflective Gaussian Splatting. In *ICLR*, 2025. 1, 5

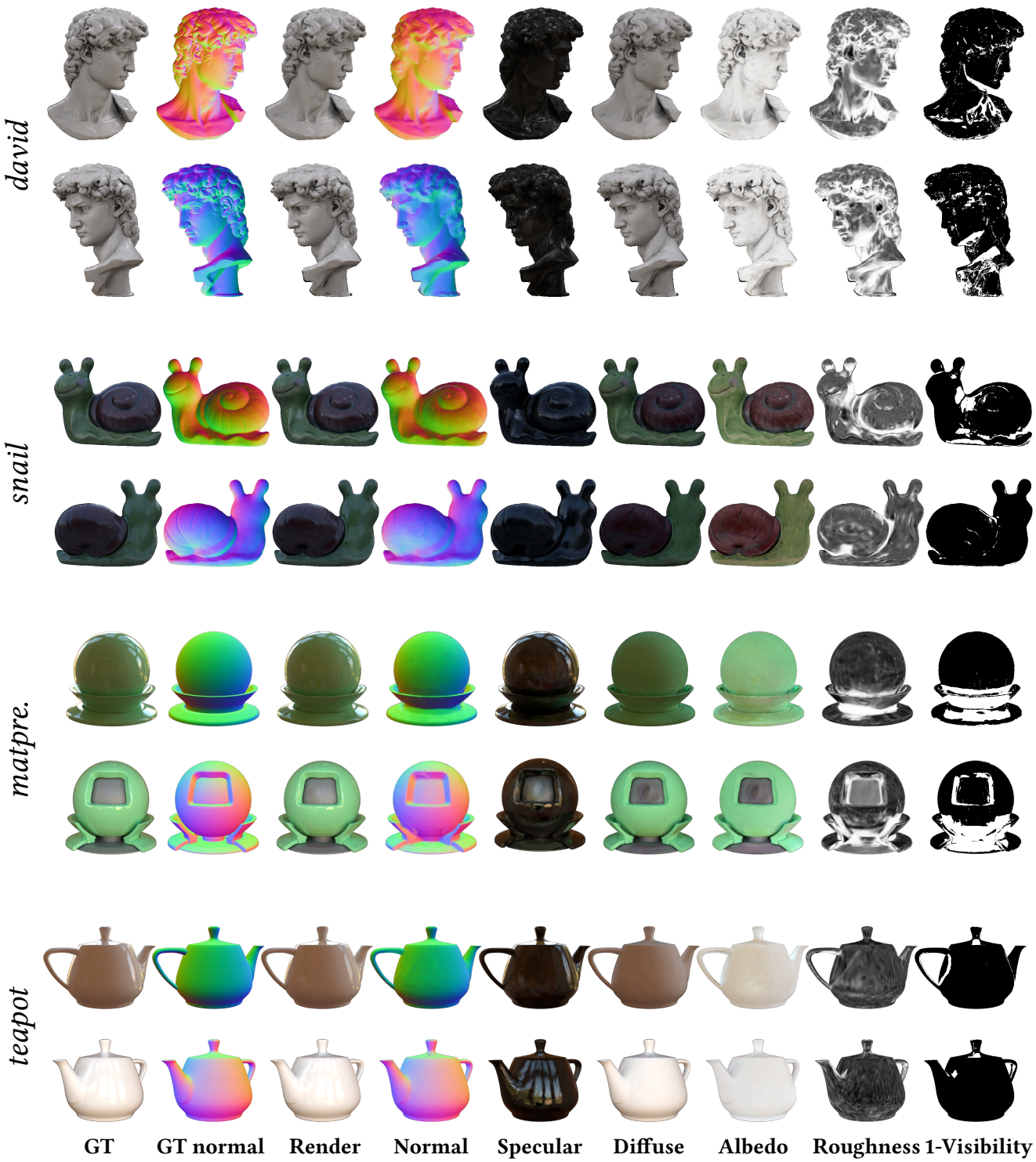

Figure 7. Additional reconstruction results with our model. From top to bottom: *david*, *snail*, *matpre.*, and *teapot*.