MAT2GS: FEEDFORWARD RELIGHTING WITH EXPLICIT MATERIALS ON 2D GAUSSIANS

Anonymous authors
Paper under double-blind review

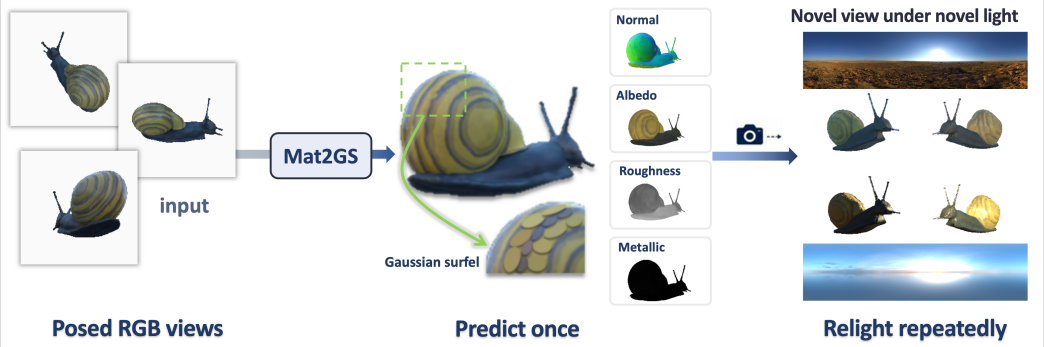

Figure 1: We propose a feedforward method that reconstructs a reusable, relightable 2D Gaussian asset from posed RGB views for novel view rendering under diverse lighting.

ABSTRACT

Relighting is essential for seamlessly integrating 3D assets into diverse environments, enabling applications such as augmented reality, simulation, and digital content creation. However, existing relighting methods often rely on implicit appearance modeling or image space material prediction coupled with deferred shading, and struggle to faithfully reproduce specular highlights and shadows under novel illumination. To mitigate this issue, we propose a feedforward relighting framework based on 2D Gaussian Splatting (2DGS) that directly attaches explicit material attributes to 2D Gaussians, establishing a unified representation of geometry and materials. Through accurate geometry and material prediction, our method produces realistic specular highlights and shadows under novel environment lighting. We further construct a large synthetic dataset by rendering a nearly 110K object subset from Objaverse, providing images with various viewpoints and illumination under corresponding material and geometry. Experiments show that Mat2GS outperforms existing open source methods in relighting quality.

1 INTRODUCTION

Relighting predicts object appearance under novel illumination, enabling captured assets to be reused across virtual environments. The task is challenging because RGB images entangle geometry, materials, illumination, visibility, and indirect light transport, while sparse views provide limited evidence for separating intrinsic materials from baked shadows and highlights.

Existing approaches broadly follow two directions. Learning based methods synthesize images under target lighting or encode appearance dependent on lighting appearance in neural representations (Srinivasan et al., 2021; Lyu et al., 2022; Jin et al., 2024; Zhang et al., 2025; Liang et al., 2025; Tang et al., 2025; Fu et al., 2026). They produce plausible results but lack explicit, geometrically aligned materials and visibility, limiting highlights and shadows that remain consistent across views. Optimization based inverse rendering explicitly decomposes geometry, reflectance, and lighting for physically based rendering (Gao et al., 2024; Liang et al., 2024; Chen et al., 2025; Gu et al., 2025),

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

but requires expensive optimization for each scene and remains sensitive to ambiguity between material and lighting. Recent feedforward inverse rendering methods (Wu et al., 2026; Dihlmann et al., 2026; Karpikova et al., 2026) motivate combining efficient generalization with explicit physical control.

To this end, we introduce **Mat2GS**, a feedforward framework with two components. First, a unified predictor of geometry and material constructs a relightable 2D Gaussian asset from posed images. It predicts geometry aligned with the surface and PBR attributes on shared anchors, keeping them aligned for reuse across novel views and lighting without refinement for each scene.

Second, an explicit PBR renderer extends the formulation inspired by IRGS (Gu et al., 2025) with a metallic and roughness model. It combines split sum environment lighting with 2D Gaussian ray tracing for visibility and single bounce indirect light, producing more faithful highlights and shadows on dielectric, glossy, and metallic surfaces. Experiments show that Mat2GS outperforms the evaluated baselines in relighting at both context and novel views, while achieving strong material recovery and competitive normal accuracy.

Our main contributions are summarized as follows:

- We construct a large synthetic relighting dataset from nearly 110K Objaverse objects, covering multiple viewpoints and environment illuminations with aligned RGB, geometry, foreground, and PBR material supervision.
- We propose a feedforward multiview framework that jointly predicts surface aligned 2D Gaussian geometry and albedo, metallic, and roughness for each Gaussian, producing a reusable explicit asset that can be relit under novel illumination without optimization for each object.
- We conduct extensive comparisons across relighting, material recovery, and novel view synthesis. The results demonstrate that Mat2GS outperforms leading open source approaches in relighting quality while maintaining geometrically and materially consistent rendering across viewpoints.

2 Related works

Existing relighting methods can be broadly grouped into implicit and explicit paradigms. Implicit approaches encode the mapping from lighting conditions to scene radiance in neural representations, avoiding an explicit decomposition of geometry, materials, and illumination. Explicit approaches instead recover these physical attributes and render the scene under novel lighting through a graphics pipeline. We review both paradigms below, with particular attention to material decomposition and prediction in explicit relighting.

2.1 Implicit relighting

Implicit relighting models appearance dependent on illumination without complete physical decomposition, using optimization for each scene, generative synthesis, or feedforward reconstruction.

Optimized neural relighting. NeRV and Neural Radiance Transfer Fields optimize neural fields for relighting and light transport (Srinivasan et al., 2021; Lyu et al., 2022). NRHints adds shadow and highlight cues without explicitly factorizing reflectance (Zeng et al., 2023). GS^3 and RNG optimize neural appearance representations on Gaussian primitives conditioned on lighting, but do not output explicit PBR maps (Bi et al., 2024; Fan et al., 2025).

Diffusion relighting. Neural Gaffer generates an object image conditioned on a target environment map (Jin et al., 2024). RGB$\leftrightarrow$X learns decomposition and synthesis with diffusion models (Zeng et al., 2024). DiffusionRenderer extends this idea to video, while SynthLight specializes diffusion relighting to portraits (Liang et al., 2025; Chaturvedi et al., 2025). These methods produce plausible lighting effects, but operate only at the image or video level.

Feedforward 3D relighting. RelitLRM and ROGR construct relightable implicit 3D representations with feedforward or distillation models (Zhang et al., 2025; Tang et al., 2025). F-RNG predicts relightable Gaussian assets from sparse views using reconstruction and intrinsic decomposition priors

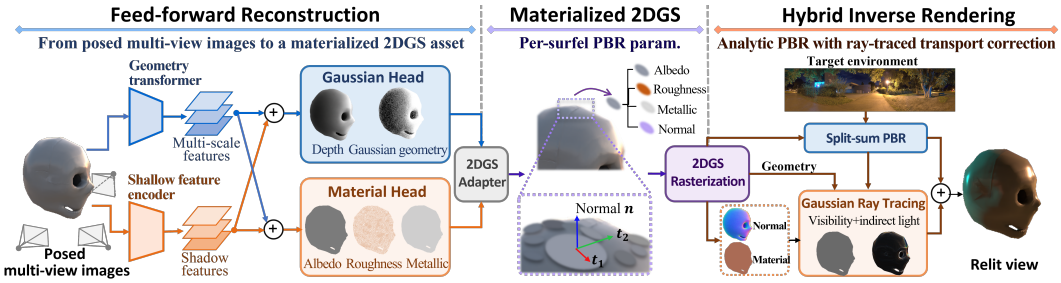

Figure 2: **Overview of Mat2GS.** Given posed multiview images, our network predicts 2D Gaussian geometry and material attributes for each surfel, forming a reusable relightable asset. Analytic split sum PBR combined with ray traced visibility and indirect illumination enables novel view relighting.

(Fu et al., 2026). Their appearance remains encoded in neural decoders or fields dependent on view direction.

Across these methods, light transport effects are learned rather than recomputed from geometry and materials, providing plausible images but weaker control under large lighting changes than explicit decomposition.

2.2 Explicit Relighting: Material Decomposition and Prediction

Explicit relighting recovers separable geometry, materials, and illumination through scene optimization, generative decomposition, or feedforward prediction.

Explicit scene optimization. Relightable 3D Gaussians, GS-IR, and GI-GS progressively introduce BRDF parameters, environment lighting, deferred shading, and path tracing into Gaussian inverse rendering (Gao et al., 2024; Liang et al., 2024; Chen et al., 2025). IRGS and IRGS++ use differentiable 2D Gaussian ray tracing for interreflection, while GHPT combines Gaussian reconstruction with hybrid path tracing (Gu et al., 2025; 2026; Bo et al., 2026). GeoSplatting adds an optimizable mesh for accurate normals and mesh based occlusion (Ye et al., 2025). These methods improve physical consistency but require optimization for each scene.

Generative decomposition. MaterialFusion and Materialist use learned priors to predict or initialize PBR properties before a physically grounded relighting stage (Litman et al., 2025; Wang et al., 2026). RGB↔X predicts intrinsic channels and lighting with diffusion models, providing an image space interface for later relighting and editing (Zeng et al., 2024). The predicted maps alone do not guarantee a complete 3D asset.

Feedforward materials. Earlier methods estimate SVBRDFs or jointly predict shape and materials from one image (Li et al., 2018; Sang & Chandraker, 2020; Vecchio et al., 2021). Material Palette and SuperMat improve PBR prediction and multiview or UV refinement (Lopes et al., 2024; Hong et al., 2025). MVInverse predicts intrinsic maps consistent across views, InvSplat attaches intrinsic attributes to structured 3D Gaussians, and ReLi3D jointly predicts geometry, PBR materials, and illumination (Wu et al., 2026; Karpikova et al., 2026; Dihlmann et al., 2026). These methods make material prediction part of explicit relighting.

Explicit methods offer stronger control over shadows, highlights, and indirect illumination, but still face optimization for each scene and incomplete or misaligned assets. Our work targets an asset ready for rendering rather than only a relit image or isolated material maps.

3 Preliminary

We introduce the explicit 2D Gaussian material representation and the rendering equation.

3.1 EXPLICIT MATERIAL REPRESENTATION

Following 2DGS and IRGS (Huang et al., 2024; Gu et al., 2025), we augment each planar Gaussian surfel with explicit PBR material attributes, the complete primitive and its geometric component are

$$\mathcal{G}_i = \{\mathcal{G}_i^{\mathrm{2D}}, \mathbf{a}_i, m_i, r_i\}, \qquad \mathcal{G}_i^{\mathrm{2D}} = \{\mathbf{p}_i, s_{u,i}, s_{v,i}, \mathbf{t}_{u,i}, \mathbf{t}_{v,i}, \alpha_i\}. \tag{1}$$

Here, $\mathcal{G}_i$ contains albedo $\mathbf{a}_i$, metallic factor m_i, roughness r_i, and geometry $\mathcal{G}_i^{\mathrm{2D}}$. The geometry stores the center, two in plane scales, tangent frame, and opacity, with $\mathbf{n}_i = \mathbf{t}_{u,i} \times \mathbf{t}_{v,i}$. Instead of view dependent spherical harmonics, albedo specifies intrinsic color, metallic factor controls dielectric to metallic reflection, and roughness controls the specular lobe width. These attributes are independent of illumination and are combined with the surface normal and incident light by the PBR renderer.

For any primitive attribute $\mathbf{F}_i$, such as depth, normal, or material, depth ordered rasterization computes the pixel value as

$$\widehat{\mathbf{F}} = \sum_{i=1}^{N} T_i \widetilde{\alpha}_i \mathbf{F}_i, \qquad \widetilde{\alpha}_i = \alpha_i G_i, \qquad T_i = \prod_{j<i} \big(1 - \widetilde{\alpha}_j\big). \tag{2}$$

Here, $G_i(\mathbf{x})$ is the projected Gaussian value, $\widetilde{\alpha}_i$ is its effective opacity, and T_i is the accumulated transmittance. The resulting attribute maps are passed to the PBR renderer.

3.2 PBR RELIGHTING

Relighting is performed in screen space. Two rasterization passes use the same depth ordered Gaussian weights to produce opacity, depth, normal, and PBR attribute maps. We normalize the rendered normal by the accumulated opacity and recover world space surface positions from camera space depth for ray tracing.

At a primary surface point $\mathbf{x}$ with view direction $\mathbf{v}$, we separate analytic environment lighting from a ray traced residual:

$$\widehat{\mathbf{L}}_{\mathrm{o}} = \mathbf{L}_{\mathrm{ana}} + \frac{1}{S} \sum_{k=1}^{S} w_k \left[\mathbf{L}_{\mathrm{ind},k} - (1 - V_k) \mathbf{L}_{\mathrm{env}}(\boldsymbol{\omega}_k) \right]. \tag{3}$$

The analytic term evaluates unshadowed environment lighting, while the residual removes blocked direct light and adds single bounce indirect illumination. For directions $\boldsymbol{\omega}_k$ sampled from density p, the importance weight is $w_k = f_r(\mathbf{x}, \mathbf{v}, \boldsymbol{\omega}_k)[\mathbf{n} \cdot \boldsymbol{\omega}_k]_+ / p(\boldsymbol{\omega}_k)$.

Using a prefiltered environment map, the analytic split sum PBR operator is

$$\begin{aligned} \mathbf{L}_{\mathrm{ana}} &= \mathcal{R}(\mathbf{a}, m, r, \mathbf{n}, \mathbf{v}) \\ &= (1 - m)\mathbf{a} \odot \mathbf{E}_{\mathrm{d}}(\mathbf{n}) + \mathbf{E}_{\mathrm{s}}(\boldsymbol{\omega}_r, r) \odot [\mathbf{F}_0 A(\mu, r) + B(\mu, r)], \\ \mu &= \mathbf{n} \cdot \mathbf{v}, \qquad \mathbf{F}_0 = 0.04(1 - m) + m\mathbf{a}, \qquad \boldsymbol{\omega}_r = \mathrm{normalize}(2\mu\mathbf{n} - \mathbf{v}). \end{aligned} \tag{4}$$

Here, $\mathbf{E}_{\mathrm{d}}$ and $\mathbf{E}_{\mathrm{s}}$ provide diffuse irradiance and roughness conditioned specular radiance. $\mathbf{F}_0$ is the reflectance at normal incidence, with 0.04 as the dielectric reference, while A and B are precomputed BRDF integration terms.

For each sampled direction, Gaussian ray tracing returns hit opacity α_k^{hit} and the corresponding surface attributes. Visibility and single bounce radiance are then

$$\begin{aligned} V_k = 1 - \alpha_k^{\mathrm{hit}}, \qquad \mathbf{v}_k^h = -\boldsymbol{\omega}_k, \\ \mathbf{L}_{\mathrm{ind},k} = \alpha_k^{\mathrm{hit}} \mathcal{R}(\mathbf{a}_k^h, m_k^h, r_k^h, \mathbf{n}_k^h, \mathbf{v}_k^h). \end{aligned} \tag{5}$$

Thus, the same PBR operator shades both the primary rasterized surface and traced hit surfaces.

Finally, the radiance is converted to sRGB and composited with the background:

$$\mathbf{I} = \mathrm{sRGB}(\widehat{\mathbf{L}}_{\mathrm{o}}) \odot \alpha + \mathbf{I}_{\mathrm{bg}}(1 - \alpha). \tag{6}$$

Here, α is the accumulated opacity and $\mathbf{I}_{\mathrm{bg}}$ is the background color.

4 APPROACH

We propose a feedforward framework that predicts an explicit, relightable 2D Gaussian asset from posed images. The predicted asset contains both surface geometry and PBR material attributes, so the same representation can be rendered under a new environment map without further optimization.

4.1 PROBLEM DEFINITION

Given a set of posed context images$\mathcal{X} = \{(\mathbf{I}_v, \mathbf{K}_v, \mathbf{T}_v)\}_{v=1}^{V}$, where $\mathbf{I}_v$ is the v-th RGB image, $\mathbf{K}_v$ is its intrinsic matrix, and $\mathbf{T}_v$ is its camera pose, our goal is to predict a explicit asset for the scene

$$\widehat{\mathcal{G}} = \{(\boldsymbol{\mu}_j, \mathbf{s}_j, \mathbf{q}_j, \alpha_j, \mathbf{a}_j, m_j, r_j)\}_{j=1}^{N}. \tag{7}$$

Here, $\boldsymbol{\mu}_j$, $\mathbf{s}_j$, and $\mathbf{q}_j$ denote the center, in plane scale, and orientation of the j-th 2D Gaussian. The scalar α_j is its opacity, while $\mathbf{a}_j$, m_j, and r_j are its albedo, metallic factor, and roughness. Following the radiance computation and alpha compositing in Eq. 6, a target view is rendered from the predicted asset. Given a target camera $(\mathbf{K}_t, \mathbf{T}_t)$ and an environment map $\mathbf{E}_t$, we write

$$\widehat{\mathbf{I}}_t = \mathcal{P}\Big(\widehat{\mathcal{G}}, \mathbf{K}_t, \mathbf{T}_t, \mathbf{E}_t\Big), \tag{8}$$

where $\mathcal{P}$ denotes depth ordered 2D Gaussian rasterization followed by PBR shading and alpha compositing. Unlike an image specific to one view, $\widehat{\mathcal{G}}$ is a reusable asset that supports unseen cameras and environment maps without geometry or material refinement.

4.2 OBJECT LEVEL DATA PREPARATION

We construct an object level dataset from approximately 110K Objaverse objects. For each object, we keep its geometry and intrinsic materials fixed while rendering it in Blender under six randomly selected environment maps. Each illumination uses 36 cameras uniformly distributed over the viewing sphere, with distances sampled from 1.5 to 2.0, providing broad angular coverage. All views are rendered at 256×256 resolution.

For each camera and light pair, we render RGB together with aligned albedo, metallic, roughness, normal, depth, and foreground mask buffers. These buffers supervise geometry and materials independent of illumination, while the mask restricts losses to valid object regions.

4.3 NETWORK FRAMEWORK

Geometry module. Our geometry module follows the feedforward multiview design of AnySplat and uses a transformer in the style of VGGT as its geometric backbone. The input views are encoded into image tokens and processed jointly so that features from different views can exchange geometric information. A depth head predicts a dense depth map and a confidence map for each context view. Using the known camera parameters, the depth maps are unprojected into 3D anchor points. A Gaussian output head then maps each anchor and its multiview feature to the opacity, in plane scale, and orientation of a 2D Gaussian aligned with the surface. The 2D Gaussian adapter normalizes the rotation and constrains the scale along the surface normal before constructing the Gaussian primitive.

2DGS material prediction head. To extend the geometric representation to explicit relighting, we add a dedicated material prediction head in parallel with the Gaussian output head. We use a DPT head to decode the multiview features and inject shallow image features during decoding to preserve local appearance details. The head predicts five values at each 3D anchor: three albedo channels, one metallic factor, and one roughness value. Sigmoid mappings constrain the material values to valid ranges. Both heads share anchors and confidence weights, keeping geometry and PBR attributes aligned during voxel aggregation and bound to the same primitives used in Eq. 4.

Relighting module. The predicted Gaussian attributes are always rasterized and passed to the PBR renderer in Sec. 3 to produce relit RGB images. Following IRGS (Gu et al., 2025), we use a PBR model parameterized by metallic factor and roughness, with split sum integration for direct environment illumination and optional Gaussian tracing for visibility and indirect transport. The renderer supports differentiation, but our main model does not backpropagate a relighting loss in image space through it; end to end renderer fine tuning is optional.

4.4 Training and Objective Functions

We train the feedforward predictor using geometry, material, and object region supervision. The default objective contains no relighting term in image space:

$$\mathcal{L} = \lambda_{\text{geo}}\mathcal{L}_{\text{geo}} + \lambda_{\text{mat}}\mathcal{L}_{\text{mat}} + \lambda_{\text{region}}\mathcal{L}_{\text{region}}. \tag{9}$$

For compactness, we denote the average of a pixelwise quantity f over a region $\mathcal{S}$ by $\langle f \rangle_{\mathcal{S}} = |\mathcal{S}|^{-1} \sum_{\mathbf{x}\in\mathcal{S}} f(\mathbf{x})$.

Geometry supervision. Let $\mathcal{P}$ be the full image, Ω the valid foreground, and $\Omega_\alpha = \{\mathbf{x} \mid \widehat{\alpha}(\mathbf{x}) > \tau\}$ the valid alpha region. The masked unit normals $\overline{\mathbf{N}}$ and $\overline{\mathbf{N}}^{\text{gt}}$ use the predicted and GT silhouettes, respectively. We group depth, normal, consistency, and distortion supervision as

$$\begin{aligned}
\mathcal{L}_{\text{geo}} &= \lambda_{\text{d}}\mathcal{L}_{\text{d}} + \lambda_{\text{n}}\mathcal{L}_{\text{n}} + \lambda_{\text{rn}}\mathcal{L}_{\text{rn}} + \lambda_{\text{cons}}\mathcal{L}_{\text{cons}} + \lambda_{\text{dist}}\mathcal{L}_{\text{dist}}, \\
\mathcal{L}_{\text{d}} &= \langle |\widehat{\mathbf{D}} - \mathbf{D}^{\text{gt}}| \rangle_{\Omega}, \qquad \mathcal{L}_{\text{n}} = \tfrac{1}{2}\langle \|\overline{\mathbf{N}} - \overline{\mathbf{N}}^{\text{gt}}\|_2^2 \rangle_{\mathcal{P}}, \\
\mathcal{L}_{\text{rn}} &= \langle 1 - \cos(\widehat{\mathbf{N}}, \mathbf{N}^{\text{surf}}) \rangle_{\Omega_\alpha}, \\
\mathcal{L}_{\text{cons}} &= \langle (\widehat{\mathbf{D}} - \mathbf{D}^{\text{vggt}})^2 \rangle_{\Omega}, \qquad \mathcal{L}_{\text{dist}} = \langle \widehat{\boldsymbol{\Delta}} \rangle_{\Omega}.
\end{aligned} \tag{10}$$

Material and object region supervision. We supervise rasterized albedo with foreground MSE and unmasked LPIPS, and use foreground MSE for roughness and metallic. The rendered alpha is matched to the binary GT mask over the full image Ω_{img}:

$$\begin{aligned}
\mathcal{L}_{\text{mat}} &= \lambda_{\text{a,mse}}\mathcal{L}_{\text{a,mse}} + \lambda_{\text{a,lpips}}\mathcal{L}_{\text{a,lpips}} + \lambda_{\text{r}}\mathcal{L}_{\text{r}} + \lambda_{\text{m}}\mathcal{L}_{\text{m}}, \\
\mathcal{L}_{\text{a,mse}} &= \langle \|\widehat{\mathbf{A}} - \mathbf{A}^{\text{gt}}\|_2^2 \rangle_{\Omega}, \qquad \mathcal{L}_{\text{a,lpips}} = \text{LPIPS}(\widehat{\mathbf{A}}, \mathbf{A}^{\text{gt}}), \\
\mathcal{L}_{\text{r}} &= \langle (\widehat{R} - R^{\text{gt}})^2 \rangle_{\Omega}, \qquad \mathcal{L}_{\text{m}} = \langle (\widehat{M} - M^{\text{gt}})^2 \rangle_{\Omega}, \\
\mathcal{L}_{\text{region}} &= \langle (\widehat{\boldsymbol{\alpha}} - \mathbf{M}^{\text{gt}})^2 \rangle_{\Omega_{\text{img}}}.
\end{aligned} \tag{11}$$

5 Experiments

We evaluate whether the predicted 2D Gaussian representation supports both physically grounded relighting and rendering consistent across views. We first describe the experimental setup, then compare our method with existing relighting and inverse rendering approaches, and finally analyze the contribution of each component through ablation studies.

5.1 Experimental Setup

Datasets. Our dataset derived from Objaverse contains approximately 110K objects, including a disjoint test split of 2K objects. We render six illuminations and 36 views per object in Blender at 256×256, using disjoint views and lighting for evaluation.

Baselines. We compare with DiffusionRenderer (Liang et al., 2025), MVInverse (Wu et al., 2026), ReLi3D (Dihlmann et al., 2026), and IRGS (Gu et al., 2025). We use the released models and official inference settings whenever available, and provide each method with the same context views and target illumination under the common evaluation protocol. Since IRGS requires computationally expensive optimization for each scene, we evaluate it on a smaller subset of the evaluation objects.

Metrics. For relit RGB images, we report PSNR, SSIM, and LPIPS. For albedo, we convert the predictions and ground truth to sRGB space and use the same three metrics. Metallic factor and roughness use RMSE, and surface normals use cosine similarity. For methods with novel view rendering, we report context and novel views separately.

Implementation details. Our model takes posed context images and predicts a reusable 2D Gaussian asset with geometry and PBR attributes. The asset is rendered at the target camera pose and environment map using the differentiable PBR renderer described in Sec. 3. All methods use identical target cameras and illumination, with eight context views and four target views in every experiment. Further training, resolution, and rendering settings are provided in App. A.

5.2 EVALUATION

Evaluation protocol. We evaluate two aspects of each method: the accuracy of its predicted material attributes and surface normals, and the quality of its relighting results. Our method predicts an explicit 3D Gaussian asset and can therefore render the four target views at novel camera poses. Under our evaluation setting, MVInverse and DiffusionRenderer do not support novel view rendering. We therefore evaluate these two methods only on context views and explicitly mark their novel view results as unavailable in Tab. 1. In addition, the released codebases of MVInverse and ReLi3D do not provide relighting functionality. We therefore reproduce their relighting procedures following the descriptions in the corresponding papers.

Material and normal prediction. We first evaluate the accuracy of the predicted intrinsic attributes. Albedo is evaluated in sRGB space using PSNR, SSIM, and LPIPS, while roughness and metallic factor are measured using RMSE. We assess surface normal accuracy using cosine similarity. For methods that support novel view rendering, we report these metrics on both context and novel views; MVInverse and DiffusionRenderer are evaluated on context views only. IRGS does not estimate a metallic factor, so its metallic results are marked as N/A in Tab. 1. Fig. 3 provides the corresponding qualitative comparison.

As shown in Tab. 1, DiffusionRenderer performs poorly on our object focused benchmark, while MVInverse and ReLi3D provide stronger material estimates. ReLi3D's mesh representation can smooth fine geometric and material details, whereas our surface aligned 2D Gaussians bind geometry and PBR attributes to the same adaptive primitives. IRGS achieves the highest normal accuracy through optimization for each scene, but its material estimates are less accurate and it does not predict metallic factor. Overall, our method provides the strongest material estimation with competitive normal accuracy in a single feedforward pass.

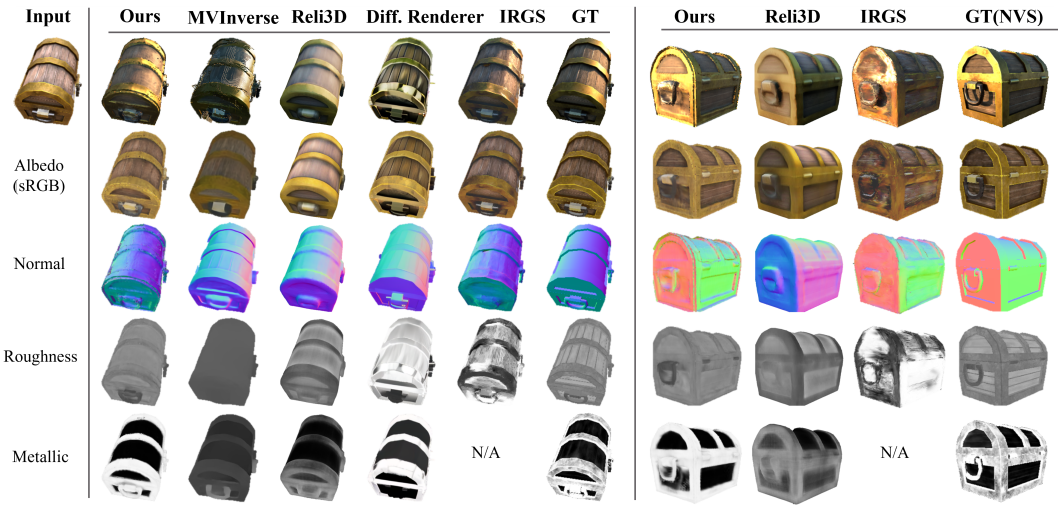

Figure 3: Qualitative comparison of predicted material attributes and surface normals. MVInverse and DiffusionRenderer are evaluated on context views because they do not support novel view rendering in our setting.

Relighting quality. We evaluate relighting under unseen target environment maps using PSNR, SSIM, LPIPS, and the qualitative results in Fig. 4. Methods supporting novel view rendering are tested at both context and novel poses, while MVInverse and DiffusionRenderer are evaluated only at context poses.

Relighting quality is closely tied to the accuracy of the predicted geometry and materials. DiffusionRenderer's inaccurate material estimates limit its relighting quality. MVInverse relies on the quality of its Pi3 point cloud and cannot effectively model occlusion of the object itself. ReLi3D provides continuous mesh surfaces, but geometry and material errors still degrade its results. IRGS is the strongest baseline due to its optimized normals, yet remains below our method because optimization for each scene struggles to disentangle geometry, materials, and illumination.

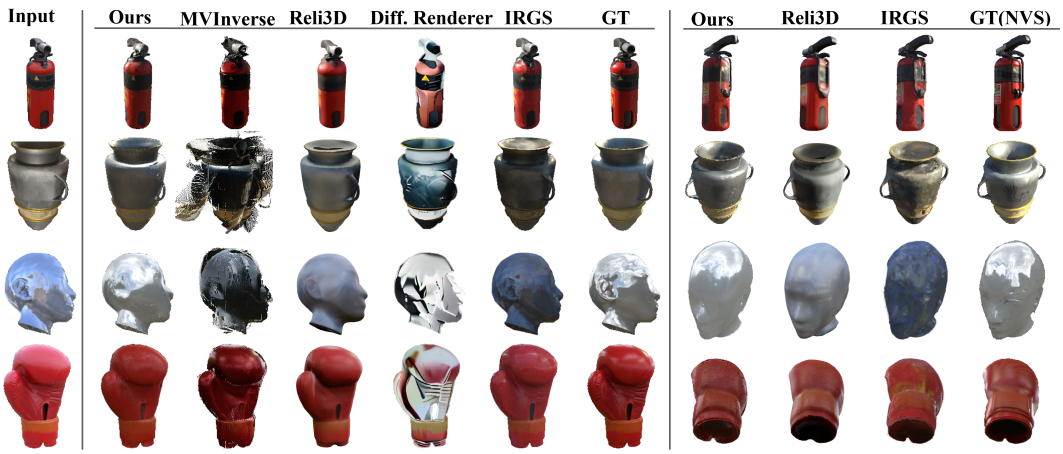

Figure 4: Qualitative relighting comparison under target environment maps. MVInverse and DiffusionRenderer are evaluated on context views, whereas methods with novel view support are additionally evaluated at novel camera poses.

Table 1: Comparison of material and normal prediction followed by relighting quality on context and novel views. Under our evaluation setting, MVInverse and DiffusionRenderer do not support novel view rendering; their reported values are therefore context view evaluations, and their novel view entries are marked as N/A. IRGS† is evaluated on a smaller subset because of its computationally expensive optimization for each scene.

PBR Attribute and Normal Estimation

Method	View	Albedo$_{sRGB}$ PSNR↑	Albedo$_{sRGB}$ SSIM↑	Albedo$_{sRGB}$ LPIPS↓	Roughness RMSE↓	Normal COS sim↑	Metallic RMSE↓
Ours	Context	**25.04**	0.8938	0.0978	**0.0606**	0.8564	**0.0791**
	Novel	**24.13**	**0.8785**	**0.1090**	**0.0656**	0.8894	**0.0869**
MVInverse	Context	24.60	**0.9001**	**0.0924**	0.1271	0.8333	0.1317
ReLi3D	Context	21.16	0.8772	0.1239	0.1237	0.8258	0.1155
	Novel	20.68	0.8667	0.1344	0.1292	0.8848	0.1211
IRGS†	Context	20.94	0.8710	0.1162	0.1873	**0.9313**	N/A
	Novel	19.76	0.8199	0.1670	0.1923	**0.9195**	N/A
DiffusionRenderer	Context	19.13	0.8624	0.1191	0.1484	0.6585	0.2062

Relighting Quality

Method	Context View PSNR↑	Context View SSIM↑	Context View LPIPS↓	Novel View PSNR↑	Novel View SSIM↑	Novel View LPIPS↓
Ours	**25.14**	**0.8944**	**0.1066**	**24.2394**	**0.8795**	**0.1178**
MVInverse	21.36	0.8331	0.1606		N/A	
ReLi3D	20.31	0.8880	0.1128	19.8859	0.8785	0.1221
IRGS†	23.42	0.8910	0.1093	21.5480	0.8375	0.1624
DiffusionRenderer	17.65	0.8326	0.1225		N/A	

5.3 Ablation Studies

We conduct two ablation studies to examine the contribution of normal supervision and explicit interaction between light and material. We remove the normal loss in the first variant and replace the explicit PBR rendering formulation with an implicit relighting model in the second.

Normal supervision. We remove the ground truth normal loss while keeping the network architecture and all other objectives unchanged. The quantitative results in Tab. 2 and qualitative comparison in Fig. 5 show that removing this loss substantially degrades the predicted normal quality and consequently reduces relighting accuracy. Surface normals determine the local incident and reflection directions in PBR shading; therefore, normal errors directly distort diffuse shading and specular

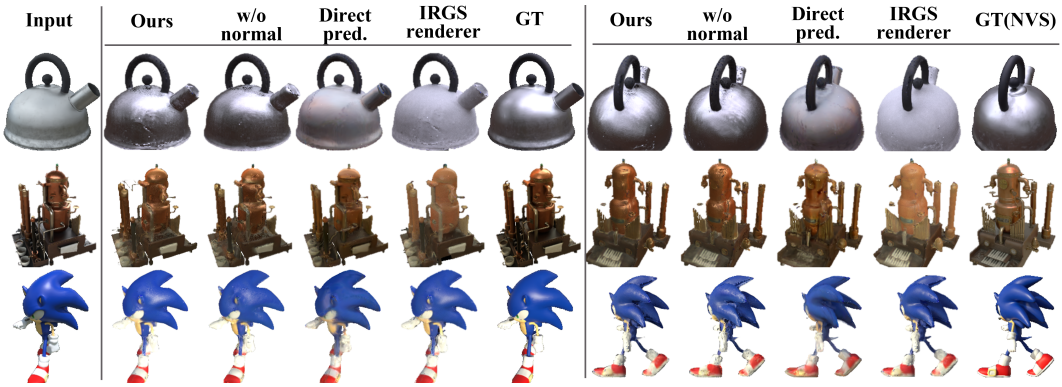

Figure 5: Qualitative comparison of the full model and variants without normal supervision, with direct relighting prediction, or with the IRGS renderer. The IRGS renderer variant combines our feedforward material predictions with the PBR renderer from IRGS.

highlights. The resulting appearance is less consistent with the target illumination, demonstrating the importance of normal supervision.

Implicit relighting. We design an implicit relighting variant that fuses the encoded target environment map with geometry transformer features and directly predicts the relit image under ground truth image supervision. As shown in Fig. 5 and Tab. 2, it produces reasonable results in many scenes, but sharp specular highlights and cast shadows remain inaccurate. These effects require precise interactions among geometry, material, visibility, and illumination. Our explicit formulation predicts material attributes independent of illumination and uses PBR to render them under the target environment map. This delegates light transport to the renderer, while the implicit variant entangles all factors and is less reliable in highlight and shadow regions.

6 Conclusion and Limitations

Conclusion. We presented Mat2GS, a feedforward framework that reconstructs explicit, relightable 2D Gaussian assets from posed multiview images. It jointly predicts surface aligned geometry and PBR attributes, enabling novel view relighting under unseen illumination without optimization for each object. Experiments show improved material, normal, and relighting results, while ablations validate normal supervision and explicit interaction between light and material.

Limitations. The training data do not cover the full diversity of real world materials, limiting the accuracy of predicted PBR attributes. Inverse rendering with 2D Gaussians also remains slower than mesh rendering, particularly for visibility and indirect illumination. Finally, feedforward Gaussian prediction can introduce geometry and opacity errors that blur rasterized results. More data and more accurate and efficient Gaussian rendering are important future work.

AI Use Statement

In this work, we used generative AI tools to design or provide feedback on research methodology or experiments, help develop theoretical models or conceptual frameworks, propose or refine hypotheses, implement methods, and interpret results. We did not use generative AI tools to generate

Table 2: Ablation study of the proposed relightable Gaussian representation and renderer. The full model predicts explicit PBR attributes and surface normals, and uses our proposed renderer.

Variant	Context View			Novel View		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Full Model	25.14	0.8944	0.1066	24.24	0.8795	0.1178
w/o Normal	24.90	0.8894	0.1103	23.99	0.8743	0.1217
Implicit Relighting	24.95	0.8978	0.1025	23.91	0.8791	0.1142

synthetic datasets, assist with translation, clean or reformat datasets, or support qualitative and thematic data analysis. Formulating mathematical claims, providing critical ingredients for proving mathematical claims, and assisting in the writing of proofs are not applicable to this work. Additionally, we used generative AI tools to summarize or analyze existing literature, discover research topics or identify gaps, brainstorm, and revise the wording and organization of the manuscript. We have reviewed all work assisted by AI. For manuscript preparation, we reviewed drafts assisted by AI through iterative revision, checking that the wording and organization reflected the intended method. We also corrected the presentation of pseudocode, citation formatting, and figure captions to maintain consistency with the manuscript and figures. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Zoubin Bi, Yixin Zeng, Chong Zeng, Fan Pei, Xiang Feng, Kun Zhou, and Hongzhi Wu. Gs^3: Efficient relighting with triple gaussian splatting. In *SIGGRAPH Asia Conference Papers*, 2024.

Jinyang Bo, Fan Dou, Wenrui Quan, Shangxun Liu, Yang Xu, Yuhe Zhang, Kang Li, and Guohua Geng. Ghpt: Real-time relightable gaussian splatting using hybrid path tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Sumit Chaturvedi, Mengwei Ren, Yannick Hold-Geoffroy, Jingyuan Liu, Julie Dorsey, and Zhixin Shu. Synthlight: Portrait relighting with diffusion model by learning to re-render synthetic faces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Hongze Chen, Zehong Lin, and Jun Zhang. Gi-gs: Global illumination decomposition on gaussian splatting for inverse rendering. In *International Conference on Learning Representations (ICLR)*, 2025.

Jan-Niklas Dihlmann, Mark Boss, Simon Donné, Andreas Engelhardt, Hendrik Lensch, and Varun Jampani. Reli3d: Relightable multi-view 3d reconstruction with disentangled illumination. In *International Conference on Learning Representations (ICLR)*, 2026.

Jiahui Fan, Fujun Luan, Jian Yang, Miloš Hašan, and Beibei Wang. Rng: Relightable neural gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Guangming Fu, Jiahui Fan, Jian Yang, Miloš Hašan, and Beibei Wang. F-rng: Feed-forward relightable neural gaussians. *arXiv preprint arXiv:2605.25975*, 2026.

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. *European Conference on Computer Vision (ECCV)*, 2024.

Chun Gu, Xiaofei Wei, Zixuan Zeng, Yuxuan Yao, and Li Zhang. Irgs: Inter-reflective gaussian splatting with 2d gaussian ray tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Chun Gu, Xiaofei Wei, Zixuan Zeng, Yuxuan Yao, and Li Zhang. Inter-reflective gaussian splatting for robust and efficient inverse rendering. *arXiv preprint arXiv:2607.22780*, 2026.

Yijia Hong, Yuan-Chen Guo, Ran Yi, Yulong Chen, Yan-Pei Cao, and Lizhuang Ma. SuperMat: Physically consistent PBR material estimation at interactive rates. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH Conference Proceedings*, 2024.

Haian Jin, Yuan Li, Fujun Luan, Yuanbo Xiangli, Sai Bi, Kai Zhang, Zexiang Xu, Jin Sun, and Noah Snavely. Neural Gaffer: Relighting any object via diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Polina Karpikova, Wenjing Bian, Haofei Xu, Hendrik Lensch, and Andreas Geiger. Invsplat: Inverse feed-forward scene splatting. *arXiv preprint arXiv:2607.02301*, 2026.

Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. Materials for Masses: SVBRDF acquisition with a single mobile phone image. In *European Conference on Computer Vision (ECCV)*, 2018.

Ruofan Liang, Zan Gojcic, Huan Ling, Jacob Munkberg, Jon Hasselgren, Zhi-Hao Lin, Jun Gao, Alexander Keller, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. DiffusionRenderer: Neural inverse and forward rendering with video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

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 (CVPR)*, 2024.

Yehonathan Litman, Or Patashnik, Kangle Deng, Aviral Agrawal, Rushikesh Zawar, Fernando De la Torre, and Shubham Tulsiani. Materialfusion: Enhancing inverse rendering with material diffusion priors. In *International Conference on 3D Vision (3DV)*, 2025.

Ivan Lopes, Fabio Pizzati, and Raoul de Charette. Material Palette: Extraction of materials from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Linjie Lyu, Ayush Tewari, Thomas Leimkühler, Marc Habermann, and Christian Theobalt. Neural radiance transfer fields for relightable novel-view synthesis with global illumination. In *European Conference on Computer Vision (ECCV)*, 2022.

Shen Sang and Manmohan Chandraker. Single-shot neural relighting and svbrdf estimation. In *European Conference on Computer Vision (ECCV)*, 2020.

Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Jiapeng Tang, Matthew Levine, Dor Verbin, Stephan J. Garbin, Matthias Nießner, Ricardo Martin-Brualla, Pratul P. Srinivasan, and Philipp Henzler. ROGR: Relightable 3D objects using generative relighting. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Giuseppe Vecchio, Simone Palazzo, and Concetto Spampinato. Surfacenet: Adversarial svbrdf estimation from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Lezhong Wang, Duc Minh Tran, Ruiqi Cui, Thomson TG, Anders Bjorholm Dahl, Siavash Arjomand Bigdeli, Jeppe Revall Frisvad, and Manmohan Chandraker. Materialist: Physically based editing using single-image inverse rendering. *International Journal of Computer Vision*, 2026.

Xiangzuo Wu, Chengwei Ren, Jun Zhou, Xiu Li, and Yuan Liu. Mvinverse: Feed-forward multiview inverse rendering in seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Kai Ye, Chong Gao, Guanbin Li, Wenzheng Chen, and Baoquan Chen. Geosplatting: Towards geometry guided gaussian splatting for physically-based inverse rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Chong Zeng, Guojun Chen, Yue Dong, Pieter Peers, Hongzhi Wu, and Xin Tong. Relighting neural radiance fields with shadow and highlight hints. In *ACM SIGGRAPH*, 2023.

Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Miloš Hašan. Rgb↔x: Image decomposition and synthesis using material- and lighting-aware diffusion models. In *ACM SIGGRAPH Conference Proceedings*, 2024.

Tianyuan Zhang, Zhengfei Kuang, Haian Jin, Zexiang Xu, Sai Bi, Hao Tan, He Zhang, Yiwei Hu, Milos Hasan, William T. Freeman, Kai Zhang, and Fujun Luan. RelitLRM: Generative relightable radiance for large reconstruction models. In *International Conference on Learning Representations (ICLR)*, 2025.

A Appendix

A.1 Detailed Training Settings

All experiments are conducted on NVIDIA A100 GPUs. We initialize the VGGT geometric backbone and depth head from pretrained weights, while the Gaussian and material prediction heads are initialized randomly. No checkpoint specific to the task is used for initialization. The complete dataset contains approximately 110K Objaverse objects, and its training split contains 70K objects. We train for 150 epochs using distributed data parallelism on 16 GPUs across two nodes, with eight processes per node. The complete training run takes approximately three days. The batch size is eight per GPU, resulting in a global batch size of 128. Each training instance contains eight context views and four target views, and the objectives for both view sets are weighted equally by 0.5.

We use AdamW with an initial learning rate of 2×10^{-4}, a weight decay of 5×10^{-3}, and momentum coefficients $(\beta_1, \beta_2) = (0.9, 0.95)$. We warm up the learning rate for 500 steps from 0.002 of its nominal value, then apply cosine decay to a minimum of 10^{-5}. The global gradient norm is clipped to 1.0, with checkpoints saved every five epochs.

For geometry supervision, we apply an L_1 loss between the rendered depth and the ground truth depth in camera space with a weight of 0.2. The rendered normal is supervised against the ground truth normal using a cosine loss with a weight of 0.05. The rendered normal consistency term, which aligns the normal of each 2D Gaussian with the rasterized normal, has a weight of 0 and is therefore disabled. We additionally enforce an MSE depth consistency loss between the rendered depth and the depth predicted by VGGT with a weight of 0.1, together with the 2DGS depth distortion regularizer weighted by 0.1.

For material and object region supervision, the albedo MSE and LPIPS terms are weighted by 1.0 and 0.05, while the roughness and metallic losses both use a weight of 0.4, and the object region opacity loss has a weight of 0.05. All active losses are enabled from the start, without relighting supervision in image space.

A.2 Evaluation Settings

All evaluations use eight context views and four target views, with identical target cameras and illumination across methods. Due to their input resolution constraints, DiffusionRenderer uses the original 256×256 inputs, while ReLi3D uses 512×512 inputs upsampled from the same 256×256 source data. All other methods are evaluated at 224×224 resolution. All metrics are computed at each method's input resolution.

A.3 PBR Rendering Settings

During evaluation, the rendered normal normalized by alpha is used for PBR shading, with the predicted metallic factor included in the BRDF. Albedo is converted from sRGB to linear RGB, while environment maps are rotated to the Blender convention and prefiltered at resolution 256 over $[0.08, 0.5]$ roughness.

Hybrid estimator derivation. Here, V, $\mathbf{L}_{\text{env}}$, and $\mathbf{L}_{\text{ind}}$ denote visibility, environment radiance, and single bounce indirect radiance. Using $V = 1-(1-V)$, the rendering equation becomes

$$\begin{aligned}\mathbf{L}_{\text{o}}(\mathbf{x},\mathbf{v}) &= \int_{\Omega} f_r\left[V\mathbf{L}_{\text{env}} + \mathbf{L}_{\text{ind}}\right][\mathbf{n}\cdot\boldsymbol{\omega}]_+ \,\mathrm{d}\boldsymbol{\omega} \\ &= \int_{\Omega} f_r\mathbf{L}_{\text{env}}[\mathbf{n}\cdot\boldsymbol{\omega}]_+ \,\mathrm{d}\boldsymbol{\omega} \\ &\quad + \int_{\Omega} f_r\left[\mathbf{L}_{\text{ind}} - (1-V)\mathbf{L}_{\text{env}}\right][\mathbf{n}\cdot\boldsymbol{\omega}]_+ \,\mathrm{d}\boldsymbol{\omega}.\end{aligned} \tag{12}$$

Eq. 4 evaluates the first term, while Monte Carlo estimation of the second gives the residual in Eq. 3, where $w_k = f_r(\mathbf{x},\mathbf{v},\boldsymbol{\omega}_k)[\mathbf{n}\cdot\boldsymbol{\omega}_k]_+/p(\boldsymbol{\omega}_k)$. Here, p is the sampling density. Gaussian ray tracing returns hit opacity and surface attributes. We set $V_k = 1-\alpha_k^{\text{hit}}$, evaluate the same split sum operator with $\mathbf{v}_k^h = -\boldsymbol{\omega}_k$, and weight the result by α_k^{hit} to obtain $\mathbf{L}_{\text{ind},k}$. Secondary visibility is enabled for

this term. Each foreground pixel uses 64 uniform hemisphere directions and 64 environment map light samples, giving 128 directions in total.

A.4 Additional Relighting Results

Fig. 6 provides additional comparisons of relighting under target environment maps. Context and novel views are shown on the left and right, respectively. Our method generally reproduces the target shading and reflections more faithfully in these examples. MVInverse's fragmented surfaces disrupt the relit appearance, while ReLi3D often produces overly smooth shading and weak specular responses. DiffusionRenderer can substantially alter the target appearance, and IRGS shows color and shading discrepancies on some objects. The novel view results further demonstrate that the same predicted 2D Gaussian asset supports relighting across camera poses, with remaining errors in fine structures and highlights.

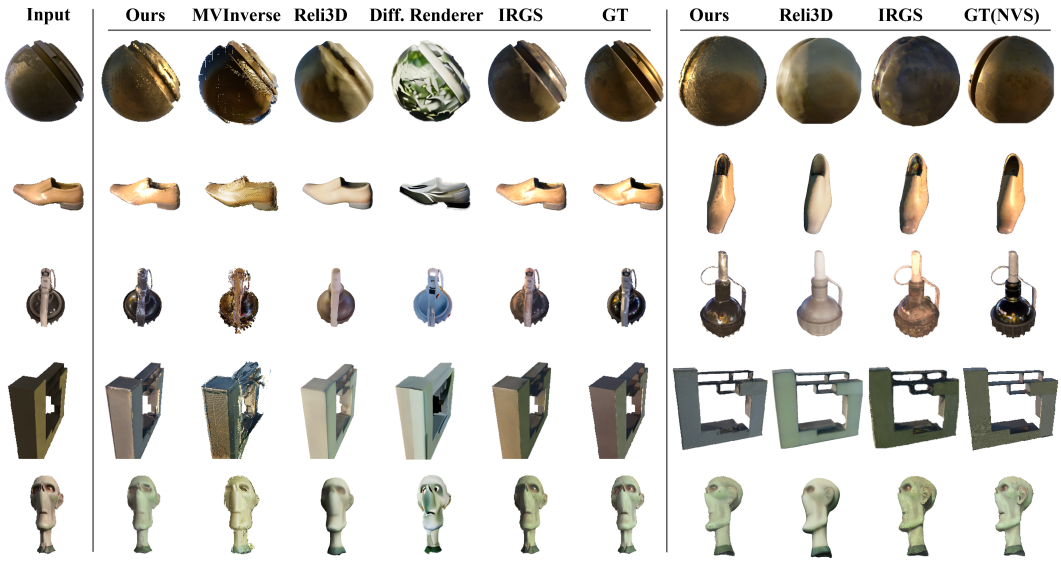

Figure 6: Additional qualitative comparisons of relighting under target environment maps. Relit images are shown at context views on the left and novel views on the right. MVInverse and DiffusionRenderer are evaluated only on context views.

A.5 Additional Material and Normal Results

To complement the main experiments, Fig. 7 provides additional qualitative results for albedo, surface normals, roughness, and metallic factor at context and novel views. Our method preserves the label texture and distinct material regions, while capturing the overall surface structure. Compared with the baselines, our predictions show less residual illumination in albedo and clearer roughness and metallic boundaries in this example. These supplementary results support the findings of the main evaluation, with the corresponding relit images illustrating how the recovered attributes determine appearance under the target illumination.

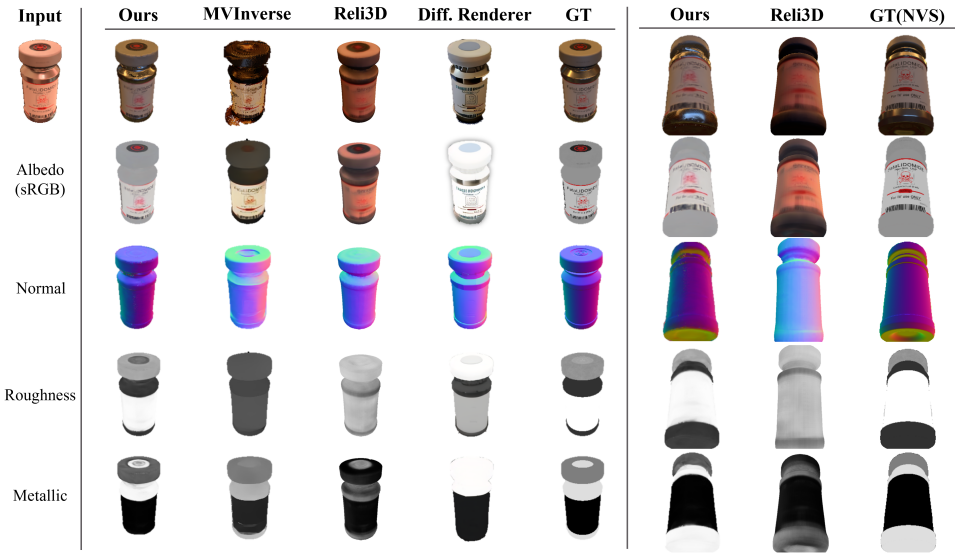

Figure 7: Additional qualitative comparison of predicted material attributes and surface normals, with corresponding relit images in the top row. Context and novel views are shown on the left and right, respectively. MVInverse and DiffusionRenderer are evaluated only on context views.