Uni-IR: One Stage is Enough for Ambiguity-Reduced Inverse Rendering

Wenhang Ge[†1], Jiawei Feng[†1], Guibao Shen[1], and Ying-Cong Chen[‡1,2]

[1]The Hong Kong University of Science and Technology (Guangzhou)
[2]The Hong Kong University of Science and Technology

Abstract

Inverse rendering aims to decompose an image into geometry, materials, and lighting. Recently, Neural Radiance Fields (NeRF) based inverse rendering has significantly advanced, bridging the gap between NeRF-based models and conventional rendering engines. Existing methods typically adopt a two-stage optimization approach, beginning with volume rendering for geometry reconstruction, followed by physically based rendering (PBR) for materials and lighting estimation. However, the inherent ambiguity between materials and lighting during PBR, along with the suboptimal nature of geometry reconstruction by volume rendering, compromises the outcomes. To address these challenges, we introduce Uni-IR, a unified framework that imposes mutual constraints to alleviate ambiguity by integrating volume rendering and physically based rendering. Specifically, we employ a physically-based volume rendering (PBVR) approach that incorporates PBR concepts into volume rendering, directly facilitating connections with materials and lighting, in addition to geometry. Both rendering methods are utilized simultaneously during optimization, imposing mutual constraints and optimizing geometry, materials, and lighting synergistically. By employing a carefully designed unified representation for both lighting and materials, Uni-IR achieves high-quality geometry reconstruction, materials, and lighting estimation across various object types.

CCS Concepts

• ***Computing methodologies*** → ***3D imaging; Reconstruction;***

1. Introduction

Multi-view 3D reconstruction is a pivotal task in computer vision and computer graphics, serving as a cornerstone for various applications such as game modeling [Gre18, LJ02], computer animation [Par12, Las87], and virtual reality [SVDSKVDM01]. Despite the remarkable progress achieved by Neural Radiance Fields (NeRF) [MST*21] and subsequent approaches like SDF-based neural implicit surface learning [WLL*21, YGKL21, OPG21], multi-view 3D reconstruction still presents challenges in bridging the gap between NeRF-based models and conventional rendering engines. Volume rendering, the core mechanism of NeRF, generates radiance without explicitly considering the interactions of materials and lighting. In contrast, conventional rendering engines derive shading through the interactions between surface materials [Nic65] and lighting. To bridge the gap, inverse rendering, the task of disentangling radiance into geometry, materials, and lighting, has garnered significant attention. This approach allows the reconstructed 3D model to be directly integrated into rendering engines, thereby playing a critical role in downstream applications such as game production [LJ02].

Recent studies [ZSH*22, SCL*23, ZLW*21, ZSD*21, LWL*23] have explored the inverse rendering task within a neural implicit surface learning framework. These approaches typically adopt a two-stage training strategy. In the initial stage, volume rendering is utilized for geometry reconstruction. Subsequently, in the second stage, physically based rendering is employed to refine materials and lighting under a fixed geometry, as shown in Figure 1 (a). However, the performance in the second stage heavily relies on the quality of geometry reconstruction in the first stage. While these methods demonstrate effectiveness on objects with diffuse materials, they often lead to suboptimal results in reconstructing reflective surfaces, consequently leading to compromised materials and lighting predictions in the second stage. More recently, several studies have focused on object reconstruction with reflections [GHZ*23, VHM*22, LWL*23, LCL*23], with some of them [LWL*23, LCL*23] further leveraging well-reconstructed geometry for inverse rendering. Despite the promising results achieved, the ill-posed nature of inverse rendering makes optimization still challenging. The lack of constraints in physically based rendering leads to suboptimal performance, with inherent ambiguity between lighting and materials, as shown in Figure 1 (b).

[†] Equal Contribution
[‡] Corresponding Author

DOI: 10.2312/pg.20251295

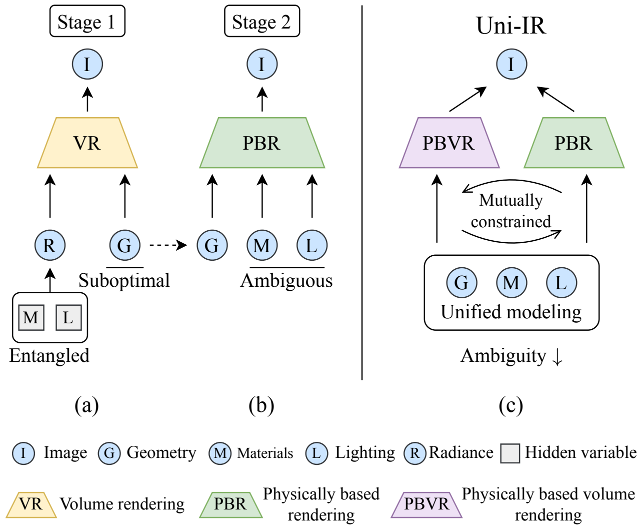

Figure 1: *Comparison between two-stage and our unified optimization framework. (a): In the first stage, VR is utilized for geometry reconstruction regardless of entangled materials and lighting. (b): In the second stage, PBR is utilized for estimating materials and lighting. However, potential suboptimal geometry and the inherent ambiguity between lighting and materials can lead to suboptimal results. (c): Our unified framework combines PBVR that takes materials and lighting into VR and PBR for imposing mutual constraints and alleviates the ambiguity.*

To tackle these challenges, we propose integrating volume rendering and physically based rendering to simultaneously optimize geometry, materials, and lighting, aiming to impose mutual constraints to alleviate ambiguity. While this idea appears straightforward, unifying the two rendering pipelines is non-trivial. Merely integrating the two into a unified framework will not introduce mutual constraints, since volume rendering lacks physical plausibility and severs the connection with physically based rendering. Conversely, the optimization becomes intractable due to the ambiguity among geometry, materials, and lighting. To address this issue, we depart from traditional volume rendering, which employs an MLP to predict radiance with entangled materials and lighting, as shown in Figure 1 (a). Instead, we integrate the principles of PBR into volume rendering, inspired by [LWL*23]. In this approach, radiance is also formulated as the interaction between materials and lighting. We refer to this method as physically based volume rendering (PBVR).

By integrating PBVR and PBR within a unified framework, we introduce a method named Uni-IR, as illustrated in Figure 1 (c). At the heart of this approach lies a meticulously crafted unified representation for materials and lighting across both rendering methods, achieving mutual constraints and optimizing geometry, materials, and lighting synergistically. In the lighting representation, we consolidate the lighting representation for both PBVR and PBR through integrated directional encoding [VHM*22]. Two distinct light MLPs are employed to model direct and indirect lighting, respectively, addressing global illumination and inter-reflection, which are crucial for differentiating indirect light from albedo and the environment map. Additionally, we parameterize the visibility term through two distinct MLPs. One is probabilistic for PBVR, while the other is deterministic for PBR, indicating whether direct or indirect lights should be used. For materials prediction, we employ a shared material MLP for both PBVR and PBR.

To summarize, our contributions are listed as follows.

- To the best of our knowledge, we introduce the first unified optimization framework for inverse rendering task, seamlessly integrating PBVR and PBR. Unifying these two rendering methods synergistically enhances the performance of inverse rendering.
- We meticulously design a unified lighting and materials representation for PBVR and PBR, effectively imposing mutual constraints and mitigating ambiguity.
- We present a comprehensive evaluation protocol, encompassing assessments of geometry reconstruction, as well as material and lighting estimation accuracy. Extensive experiments conducted on multiple datasets demonstrate the effectiveness of the proposed framework.

2. Related Work

2.1. Multi-view 3D reconstruction

Traditional multi-view 3D reconstruction typically employs Multi-View Stereo (MVS) techniques [CL96, FL95, FP09, SZFP16, SF16, XT19], with the objective of reconstructing scene geometry from multi-view images. These techniques utilize multi-view consistency to establish correspondences and estimate depth values across different views, yielding a point cloud reconstruction. However, MVS methods encounter challenges in reconstructing reliable geometry in specific scenarios, such as surfaces with specular reflections and regions with low texture.

With the recent advancements in deep learning, learning-based approaches utilizing implicit surface representations have emerged, where neural networks are employed to map continuous points to either an occupancy field [MON*19, PNM*20] or a Signed Distance Function (SDF) [PFS*19]. Unlike traditional methods, these approaches are immune to appearance changes as they rely on 3D ground truth supervision. However, these methods typically necessitate additional supervision corresponding to the occupancy value or SDF for each point. Unfortunately, such supervision may not always be readily available when utilizing solely multi-view 2D images, thereby restricting their scalability.

The advent of volumetric approaches in NeRF [MST*21] has sparked significant interest in 3D reconstruction utilizing neural implicit surface representations [OPG21, YGKL21, WLL*21]. Subsequent research endeavors have continued to enhance the reconstruction performance across various aspects. Despite demonstrating promising performance in 3D reconstruction, these methods still struggle to accurately recover the geometry of specular surfaces. Therefore, in this study, we concentrate on the reconstruction of reflective objects, along with the estimation of material and lighting properties.

2.2. Modeling for Object with Reflection

Recently, several studies [BBJ*21, SDZ*21, ZLW*21, ZSD*21, VHM*22, GHZ*23, LWL*23] have focused on modeling objects with reflection. Some studies [VHM*22, BBJ*21, ZSD*21, RPHD20, KLR*22, LYL*23] tackle rendering tasks, where they model view-dependent reflective appearances by decomposing a scene into shape, reflectance, and illumination for novel view synthesis and relighting. Other studies [GHZ*23, LWL*23, WHZL24, FSV*23, LCL*23] focus on reconstructing reflective geometry by modeling specular light more reasonably or mitigating the effects of specular surfaces. For example, Ref-NeuS [GHZ*23] and NeP [WHZL24] reduce the impact of highly uncertain reflective regions while enhancing the significance of less altered areas. Additionally, ENVIDR [LCL*23] and NeRO [LWL*23] adopt a more physically plausible approach to model specular color, leading to notable improvements in reconstruction performance. Typically, physically based rendering is employed to further estimate materials and lighting given well-reconstructed geometry. Despite achieving promising results, the entanglement between illumination and materials compromises outcomes due to the ill-posed nature of inverse rendering, inevitably leading to suboptimal results in many cases. In this study, we propose a unified framework that imposes mutual constraints and alleviates the entanglement issue.

2.3. Inverse Rendering by Neural Implicit Learning

Inverse rendering [BM14, NDVZJ19] aims to decompose image appearance into intrinsic properties such as geometry, materials, and lighting. This task has posed a challenge in computer vision and graphics due to its ill-posed nature. Recovering reliable intrinsic properties is particularly difficult because of the limited constraints added during optimization. To address this challenge, most existing methods [ZSH*22,LWL*23,YDMH99,ZSD*21,YZL*22, WHZL24] employ a geometry-first optimization framework. Initially, they utilize volume rendering to reconstruct geometry. Subsequently, in the second stage, physically based rendering is used for materials and lighting estimation. However, ambiguity between materials and lighting hinders the second-stage optimization. These methods do not establish a connection between volume rendering and physically based rendering, as they are performed separately in two distinct stages. Our approach integrates PBVR, which incorporates the concept of PBR into traditional volume rendering to establish a connection with PBR, and PBR within a unified framework. Featuring a carefully designed unified representation for both lighting and material representation, our method effectively imposes mutual constraints and mitigates ambiguity.

3. Method

With N calibrated multi-view images denoted as $\mathcal{X} = \{\mathbf{I}_i\}_{i=1}^{N}$, our objective is to address the inverse rendering that simultaneous reconstructing of the object's geometry and estimating the materials and lighting. We commence by providing a succinct overview of volume rendering and physically based rendering in Section 3.1. Next, we introduce physically based volume rendering (PBVR), and how we integrate PBR and PBVR into a unified framework in Section 3.2. Subsequently, we delve into the design of unified lighting and materials representations in Section 3.3. Lastly, Section 3.4 presents full optimization. An overview of our framework is illustrated in Figure 2.

3.1. Preliminaries

Volume Rendering. Volume rendering [KVH84] used in NeRF [MST*21] aims at multi-view 3D reconstruction and novel view synthesis. The core idea is to represent the continuous attributes (i.e., density and radiance) of a 3D scene with neural networks. α compositing [Max95] aggregates these attributes along a ray $\mathbf{r}$ to approximate the pixel RGB values:

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{i=1}^{P} T_i \alpha_i \mathbf{c}_i, \tag{1}$$

where $T_i = \exp\left(-\sum_{j=1}^{i-1} \alpha_j \delta_j\right)$ and $\alpha_i = 1 - \exp(-\sigma_i \delta_i)$ denote the transmittance and alpha value of sampled point, respectively. δ_i is the distance between neighboring sampled points. P is the number of sampled points along a ray. σ_i and $\mathbf{c}_i$ are predicted attributes by the neural networks. The training object $\mathcal{L}$ is the mean square error between the ground-truth pixel color $\mathbf{C}(\mathbf{r})$ and the rendered color $\hat{\mathbf{C}}(\mathbf{r})$ formulated as

$$\mathcal{L}_{\text{render}} = \sum_{\mathbf{r}\in\mathcal{R}} \|\mathbf{C}(\mathbf{r}) - \hat{\mathbf{C}}(\mathbf{r})\|_2^2, \tag{2}$$

where $\mathcal{R}$ is the set of all rays shooting from the camera center to image pixels. Subsequent approaches use Signed Distance Function (SDF) instead opacity σ to define the geometry. Following NeuS [WLL*21], the formulation of α_i is calculated from the signed distance rather than density σ_i as

$$\alpha_i = \max\left(\frac{\Phi_s(g(\mathbf{x}_i)) - \Phi_s(g(\mathbf{x}_{i+1}))}{\Phi_s(g(\mathbf{x}_i))}, 0\right), \tag{3}$$

where g is the geometry network, which maps a position $\mathbf{x}$ to its signed distance $g(\mathbf{x})$. $\Phi_s(x) = \left(1 + e^{-sx}\right)^{-1}$ and $1/s$ is a trainable parameter which indicates the standard deviation of $\Phi_s(x)$.

Physically Based Rendering. Physically based rendering aims to produce photo-realistic 2D images given geometry, materials and lighting. At its core, the rendering equation [Kaj86] models the interaction between materials and lighting in a physically plausible manner. It inherently represents an integral equation that describes the equilibrium of light in a scene. The formula is expressed as

$$c(\mathbf{x}, \omega_\mathbf{o}) = \int_\Omega f(\mathbf{x}, \omega_\mathbf{o}, \omega_\mathbf{i}) L_i(\mathbf{x}, \omega_\mathbf{i})(\omega_\mathbf{i} \cdot \mathbf{n}) d\omega_\mathbf{i}, \tag{4}$$

where $\omega_\mathbf{o}$ is the viewing direction of the outgoing light, L_i is the incident light of direction $\omega_\mathbf{i}$ sampled from the upper hemisphere Ω of the surface point $\mathbf{x}$, and $\mathbf{n}$ is the surface normal. f is the BRDF properties. The function f consists of a diffused and a specular component

$$f(\mathbf{x}, \omega_\mathbf{o}, \omega_\mathbf{i}) = (1 - m)\frac{\mathbf{a}}{\pi} + \frac{DFG}{4(\omega_\mathbf{i} \times \mathbf{n})(\omega_\mathbf{o} \times \mathbf{n})}, \tag{5}$$

where $m \in [0,1]$ is the metallic of the surface point. $\mathbf{a} \in [0,1]^3$ is the albedo color of the point. D is the normal distribution function, F is the Fresnel term and G is the geometry term, which are all determined by the metallic m, the roughness r and the albedo $\mathbf{a}$. We

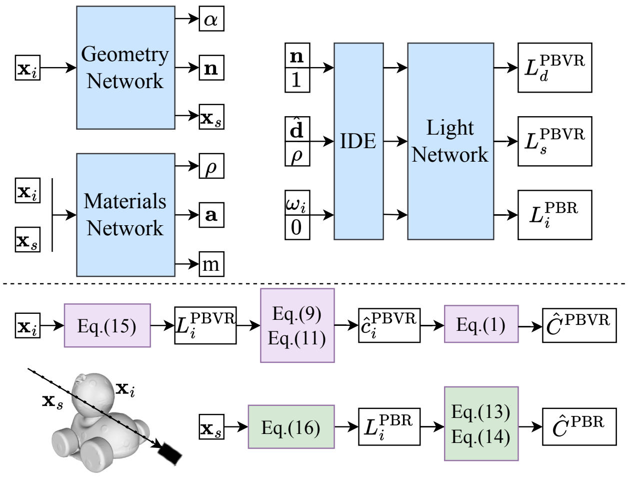

Figure 2: *Overall of our framework. Geometry network takes ray-based points $\mathbf{x}_i$ and outputs α, $\mathbf{n}$, based on which we compute surface point $\mathbf{x}_s$. Materials network predicts roughness ρ, metallic m and albedo $\mathbf{a}$ for both ray-based points and surface-based points. Given normal $\mathbf{n}$, roughness $\rho = 1$ and reflective direction $\hat{\mathbf{d}}$, predicted ρ for $\mathbf{x}_i$, light network predicts diffuse and specular lighting as Eq.14, respectively. Given sampled direction ω_i and $\rho = 0$ for $\mathbf{x}_s$, light network predicts ambient lighting as Eq.15.*

detail the expression of D, F and G in the Supplement. With Eq. 4 and Eq. 5, the outgoing radiance is given by

$$\mathbf{c}(\mathbf{x}, \omega_{\mathbf{o}}) = \mathbf{c}_{\text{d}}(\mathbf{x}, \omega_o) + \mathbf{c}_{\text{s}}(\mathbf{x}, \omega_o), \tag{6}$$

$$\mathbf{c}_{\text{d}}(\mathbf{x}, \omega_{\mathbf{o}}) = (1-m)\mathbf{a} \int_{\Omega} L_i(\mathbf{x}, \omega_{\mathbf{i}}) \frac{(\omega_{\mathbf{i}} \cdot \mathbf{n})}{\pi} d\omega_{\mathbf{i}}, \tag{7}$$

$$\mathbf{c}_{\text{s}}(\mathbf{x}, \omega_{\mathbf{o}}) = \int_{\Omega} \frac{DFG}{4(\omega_{\mathbf{i}} \times \mathbf{n})(\omega_{\mathbf{o}} \times \mathbf{n})} L_i(\mathbf{x}, \omega_{\mathbf{i}})(\omega_{\mathbf{i}} \cdot \mathbf{n}) d\omega_{\mathbf{i}}. \tag{8}$$

3.2. Unifying Volume Rendering and Physically Based Rendering

It is not trivial to integrate volume rendering (VR) and physically based rendering (PBR) into a unified framework for simultaneous reconstructing the object's geometry and estimating its materials and lighting. A naive solution is to evaluate the rendering equation on surface points. However, materials and lighting remain only related to PBR without integrating with volume rendering, thus failing to introduce mutual constraints, since traditional volume rendering adopts an MLP for direct radiance prediction, entangling the materials and lighting. Consequently, the optimization process becomes intractable due to the ambiguity among geometry, materials, and lighting.

To guarantee that volume rendering also incorporates materials and lighting, we integrate it with the principles of PBR, which allows the radiance to be computed by modeling the interaction between materials and lighting. Inspired by NeRO [LWL*23], we represent the radiance of each sampled point along a ray using a simplified rendering equation, which approximates the lighting with light MLP instead of integral in Eq. 7 and 8, termed PBVR. The diffuse and specular components are

$$\begin{aligned} \mathbf{c}_{\text{d}}^{\text{PBVR}}(\mathbf{x}, \omega_{\mathbf{o}}) &= (1-m)\mathbf{a}L_{\text{d}}^{\text{PBVR}}, \\ L_{\text{d}}^{\text{PBVR}} &\approx \int_{\Omega} L_i(\mathbf{x}, \omega_{\mathbf{i}}) D(\mathbf{n}, 1) d\omega_{\mathbf{i}}, \end{aligned} \tag{9}$$

$$\begin{aligned} \mathbf{c}_{\text{s}}^{\text{PBVR}}(\mathbf{x}, \omega_{\mathbf{o}}) &= m\mathbf{a}L_{\text{s}}^{\text{PBVR}}, \\ L_{\text{s}}^{\text{PBVR}} &\approx \int_{\Omega} L_i(\mathbf{x}, \omega_{\mathbf{i}}) D(\hat{\mathbf{d}}, \rho) d\omega_{\mathbf{i}}, \end{aligned} \tag{10}$$

where $L_{\text{d}}^{\text{PBVR}}$, $L_{\text{s}}^{\text{PBVR}}$ are the approximated diffuse and specular light. $D(\hat{\mathbf{d}}, \rho)$ is the normal distribution function (i.e., specular lobe), $\hat{\mathbf{d}}$ is the reflective direction. $D(\mathbf{n}, 1) \approx \frac{(\omega_{\mathbf{i}} \cdot \mathbf{n})}{\pi}$ is the diffuse lobe. We elaborate the simplified process from Eq. 8 to Eq. 10 in the Supplement.

For PBR, we evaluate the rendering equation at predicted surface points, which can be localized by integrating the depth values of sampled points along a ray as

$$\mathbf{x}_s = \mathbf{o} + \mathbf{d} \sum_{i=1}^{P} T_i \alpha_i t_i, \tag{11}$$

where $\mathbf{o}$ is the camera origin, $\mathbf{d}$ is the camera direction and t_i is the depth of i-th sampled point. We adopt Monte Carlo sampling to approximate the diffuse color and specular color. The diffused color is estimated by sampling N_d rays with a cosine-weighted probability

$$\mathbf{c}_{\text{d}}^{\text{PBR}}(\mathbf{x}_s, \omega_{\mathbf{o}}) = (1-m)\mathbf{a} \sum_{i}^{N_d} L_i^{\text{PBR}}, \tag{12}$$

where i indicates the i-th sampled direction. For specular color, we adopt the GGX distribution as normal distribution D. We sample N_s rays follows DDX distribution [CT82] to estimate specular color

$$\mathbf{c}_{\text{s}}^{\text{PBR}}(\mathbf{x}_s, \omega_{\mathbf{o}}) = \frac{1}{N_s} \sum_{i}^{N_s} \frac{FG(\omega_{\mathbf{o}} \cdot \mathbf{h})}{(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \omega_{\mathbf{o}})} L_i^{\text{PBR}}, \tag{13}$$

where $\mathbf{h}$ is the half-way vector between $\omega_{\mathbf{i}}$ and $\omega_{\mathbf{o}}$, L_i^{PBR} is predicted light of i-th sampled direction.

Both rendering methods require materials and lighting for shading. By integrating PBVR and PBR with a carefully crafted materials and lighting representation, we enhance the mutual constraints for inverse rendering, thereby reducing the probability of converging to suboptimal results.

3.3. Lighting and Materials Representation

All the radiance terms in Eq. 9, Eq. 10, Eq. 12, and Eq. 13 depend on lighting and materials. Appropriately representing these elements is essential for effectively imposing mutual constrains and mitigating ambiguity among geometry, materials, and lighting. **Lighting Representation.** Given the crucial role that global illumination and inter-reflection play in distinguishing indirect light from albedo and environment maps, we utilize two distinct MLPs to separately encode direct and indirect lighting. The direct light MLP $l_{\text{direct}}(SH(\omega_i))$ takes only direction as input, ensuring a globally consistent direct environment map. $SH(\cdot)$ is the directional encoding using spherical harmonics as basis functions. This model is applicable when the path from point $\mathbf{x}$ to direction ω_i is unobstructed. In contrast, the indirect light MLP $l_{\text{indirect}}(SH(\omega_i), \mathbf{x})$ requires both position and direction as input to accommodate the spatial variability of indirect lighting across the scene. This model is used when the path from $\mathbf{x}$ to ω_i encounters obstructions.

To establish a unified lighting representation, we utilize integrated directional encoding (IDE) [VHM*22], which shows the integral of light in Eq. 9 and Eq. 10 has a closed-form solution by representing the $L_i(\mathbf{x}, \omega_i)$ with spherical harmonics, based on direction and roughness denoted as $\text{IDE}(\omega, \text{k})$. Although the roughness term in $\text{IDE}(\omega, \rho)$ is defined by the von Mises-Fisher (vMF) distribution, which differs from the roughness term in the GGX distribution used in PBR, both serve similar functions by defining positively correlated concentration. We optimize the roughness as the parameter in the GGX distribution and use it for lighting approximation in PBVR.

For PBVR, the integrals of diffuse and specualr light can be approximated by

$$L_{\text{d}}^{\text{PBVR}} = l_{\text{direct}}\left(\text{IDE}(\mathbf{n}, 1)\right), \quad L_{\text{s}}^{\text{PBVR}} = l_{\text{direct}}(\text{IDE}(\hat{\mathbf{d}}, \rho)). \quad (14)$$

In PBR, the diffuse and specular light are both computed by

$$L_i^{\text{PBR}} = l_{\text{direct}}\left(\text{IDE}(\omega_i, 0)\right), \quad (15)$$

where ρ is set to 0 since sampled directions are deterministic instead of a distribution. For indirect light, the position $\mathbf{x}$ is additionally inputted. We consolidate the lighting representation for both PBVR and PBR, encompassing both specular and diffuse components. This consolidation effectively imposes constraints on lighting optimization and alleviates entanglement issues.

Visibility Representation. Given the inclusion of indirect light in our lighting representation, it is critical to estimate a visibility term to correctly apply direct or indirect light. In PBR, light is determined through Monte Carlo sampling, where each sampled direction is deterministic, resulting in binary visibility values of either 0 or 1, denoted as $v_i^{\text{PBR}} \in \{0, 1\}$. An MLP maps the surface point $\mathbf{x}_s$ and sampled direction ω_i to visibility, defined as $v_i^{\text{PBR}} = \mathcal{V}^{\text{PBR}}(\mathbf{x}_s, \omega_i)$. In PBVR, visibility is probabilistic, denoted as $v^{\text{PBVR}} \in [0, 1]$, since lighting representation in Eq. 14 approximates the specular light using a single direction and roughness. When roughness is large, the light integral is influenced not only by the reflective direction $\hat{\mathbf{d}}$. Thus, another MLP maps the sampled point $\mathbf{x}$ and $\text{IDE}(\hat{\mathbf{d}}, \rho))$ to visibility, denoted as $v^{\text{PBVR}} = \mathcal{V}^{\text{PBVR}}(\mathbf{x}, \text{IDE}(\hat{\mathbf{d}}, \rho))$. To account for the deterministic and probabilistic property, we use visibility by ray-marching in geometry network and visible proportion by Monte Carlo sampled directions as supervision, respectively. The visibility loss is given by

$$\mathcal{L}_{vis} = \|v_i^{\text{PBR}} - v_i^{\text{march}}\|_1 + \|v^{\text{PBVR}} - \frac{1}{N_s}\sum_{i=1}^{N_s} v_i^{\text{PBR}}\|_1. \quad (16)$$

Given the visibility, the light $L_{\text{s}}^{\text{PBVR}}$ in Eq. 14 can be expressed as

$$\begin{aligned} L_{\text{s}}^{\text{PBVR}} &= v^{\text{PBVR}} l_{\text{direct}}(\text{IDE}(\hat{\mathbf{d}}, \rho)) \\ &+ (1 - v^{\text{PBVR}}) l_{\text{indirect}}(\text{IDE}(\hat{\mathbf{d}}, \rho), \mathbf{x}). \end{aligned} \quad (17)$$

Since diffuse light primarily contains low-frequency information, we do not explicitly model the indirect diffused light. The light L_i^{PBR} in Eq. 15 is modified as

$$L_i^{\text{PBR}} = v_i^{\text{PBR}} l_{\text{direct}}(\text{IDE}(\omega_i, 0)) + (1 - v_i^{\text{PBR}}) l_{\text{indirect}}(\text{IDE}(\omega_i, 0), \mathbf{x}_s). \quad (18)$$

Materials Representation. Material representation, including metallic m, roughness ρ, and albedo $\mathbf{a}$, is conducted using a material MLP $\mathcal{M}_{\text{material}}$ based on position $\mathbf{x}$, denoted as $\{m, \rho, \mathbf{a}\} = \mathcal{M}_{\text{material}}(\mathbf{x})$, and these predictions are shared across PBR-based rendering and PBR. The distinction lies in the fact that material prediction operates on ray-based points and surface-based points, respectively. This difference introduces two types of constraints for material optimization.

3.4. Optimizing

During the training process, our total loss function is

$$\mathcal{L} = \mathcal{L}_{\text{render}}^{\text{PBVR}} + \lambda_{\text{PBR}} \mathcal{L}_{\text{render}}^{\text{PBR}} + \lambda_{\text{eik}} \mathcal{L}_{\text{eik}} + \lambda_{\text{vis}} \mathcal{L}_{\text{vis}} + \lambda_{\text{mat\_reg}} \mathcal{L}_{\text{mat\_reg}}, \quad (19)$$

where $\mathcal{L}_{\text{render}}$ is the Charbonier loss [BMV*22] calculated between the rendered color and the ground-truth color. In PBVR, the rendered color is derived from Eq. 1, where each $\mathbf{c}_i$ combines $\mathbf{c}_{\text{d}}^{\text{PBVR}}$ and $\mathbf{c}_{\text{s}}^{\text{PBVR}}$ as outlined in Eqs. 9 and 10, and α_i determined by Eq. 3. In the context of PBR, the rendered color is formulated as $\mathbf{C} = \mathbf{c}_{\text{d}}^{\text{PBR}} + \mathbf{c}_{\text{s}}^{\text{PBR}}$, based on Eq. 12 and Eq. 13. $\mathcal{L}_{\text{eik}}$ is an eikonal term [GYH*20] to regularize the gradients of geometry network formualated as

$$\mathcal{L}_{\text{eik}} = \frac{1}{P}\sum_{i=1}^{P}\left(|\nabla f(\mathbf{x})| - 1\right)^2. \quad (20)$$

$\mathcal{L}_{\text{mat\_reg}}$ is a smoothness regularization to ensure the material more smooth in the space

$$\mathcal{L}_{\text{mat\_reg}} = \|\mathcal{M}(\mathbf{x}_s) - \mathcal{M}(\mathbf{x}_s + \epsilon)\|_2, \quad (21)$$

where $\epsilon = 5e-3$.

4. Experiments

4.1. Datasets and Evaluation Protocol

To evaluate the effectiveness of our method, we conducted experiments on objects from several datasets. These include synthetic data from ShinyBlender [VHM*22] and CompoBlender, where objects are composed from ShinyBlender or Blender [MST*21], featuring more complex scenes and inter-reflections (see Supplement

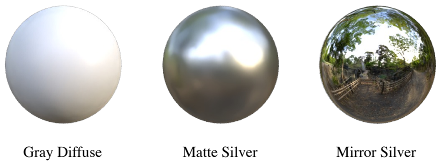

Figure 3: *Three spheres with different materials: mirror silver, matte silver, and diffuse grey*

Proceedings published by Eurographics - The European Association for Computer Graphics.

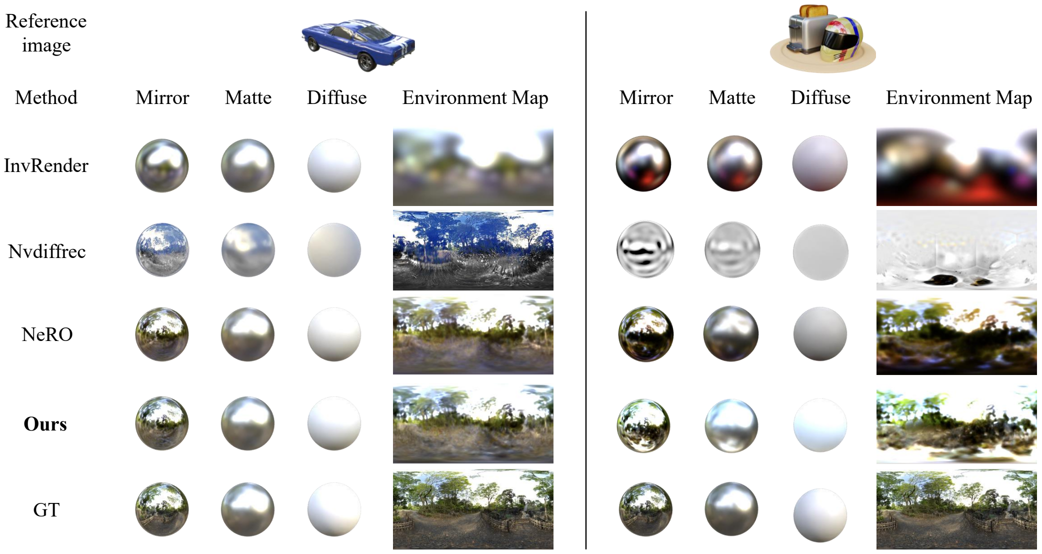

Figure 4: *The environment map and rendered spheres on "car" and "toahel". We ran InvRender, Nvidiffrec, NeRO official implementations. Our method obviously produces better result.*

Table 1: *Comparison with state-of-the-art methods on ShinyBlender and CompoBlender Dataset. **Bold** results have the best score. Our method outperforms these methods by a large margin.*

Method	ShinyBlender			CompoBlender		
	Geometry	Materials	Lighting	Geometry	Materials	Lighting
GShader	1.37	0.041 / 0.069 / 20.10	13.71 / 2.24 / 2.09	1.69	0.125 / 0.101 / 16.01	12.35 / 2.21 / 1.52
Nvdiffrec	2.59	0.045 / 0.074 / 19.90	14.11 / 2.30 / 2.05	2.95	0.138 / 0.110 / 15.93	10.70 / 2.12 / 1.41
InvRender	1.39	0.035 / - / -	11.38 / 2.09 / 2.04	1.35	0.069 / - / -	15.09 / 2.40 / 1.90
NeRO	0.67	0.023 / 0.030 / 22.26	8.86 / 1.65 / 1.97	2.05	0.063 / 0.055 / 17.68	9.40 / 1.73 / 1.04
Ours	**0.58**	**0.015 / 0.025 / 23.21**	**7.91 / 1.58 / 1.45**	**0.82**	**0.039 / 0.026 / 18.84**	**8.17 / 1.54 / 0.88**

for more details), as well as real captured data from Stanford-ORB [KZY*24].

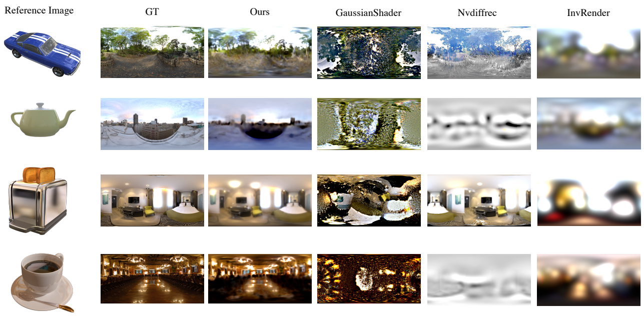

Figure 5: *Visualization of estimated environment map. Our method can recover fine-grained environment map given only multi-view 2D images.*

ShinyBlender The ShinyBlender dataset, introduced in Ref-NeRF [VHM*22], aims at the novel view synthesis task for specular surfaces. The original dataset does not include ground truth for diffuse albedo, metallic, and roughness. We re-rendered the dataset using Blender, maintaining consistent camera poses with the original dataset for each object.

CompoBlender We combined individual objects from the ShinyBlender and Blender datasets to create the CompoBlender dataset. This dataset is designed to validate the effectiveness of our method in more complex scenes. First, we combined the "helmet" from ShinyBlender with a part of the "hotdog" from the Blender dataset to create the "hothel" dataset, which features both shiny and diffuse materials. Second, we combined the "toaster" and "helmet" from ShinyBlender to create the "toahel" dataset, which includes indirect lighting and inter-reflections. For rendering multi-view images, we implemented the code from NeRFactor [ZSD*21]. Additionally, we added output nodes for metallic, roughness, and diffuse

Table 2: *Comparison with cutting-edge method NeRO on Stanford-ORB Dataset. **Bold** results have the best score. Our method performs better on real captured dataset.*

Method	Geometry	Relighting			Material
	CD ↓	PNSR ↑	SSIM ↑	LPIPS ↓	PSNR ↑
NeRO	1.35	25.45	0.898	0.054	23.25
Ours	**0.97**	**26.13**	**0.902**	**0.051**	**24.84**

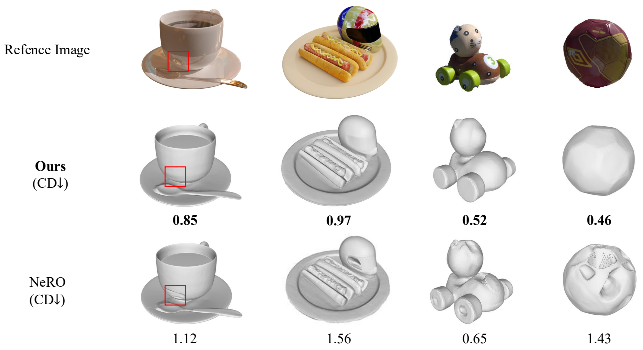

Figure 6: *The qualitative comparison of reconstruction between NeRO and our method. Incorrect material estimation hinders the geometry reconstruction.*

albedo to evaluate materials. For the environment map, we used the same environment map as “musclecar” for these two scenes.

Stanford-ORB The dataset comprises 14 common objects with different materials captured in 7 natural scenes. For each object, 60 training views and 10 testing views are provided, featuring both high dynamic range (HDR) and low dynamic range (LDR) images under three different scenes. We used LDR images for training and testing. For each object, one scene is selected for training, while the remaining two scenes are used for relighting evaluation. Specifically, we observed that there is always one scene where the object is captured in an outdoor environment. This outdoor scene was consistently chosen for training.

We present a comprehensive evaluation protocol, encompassing assessments of geometry reconstruction accuracy as well as materials and lighting estimation accuracy.

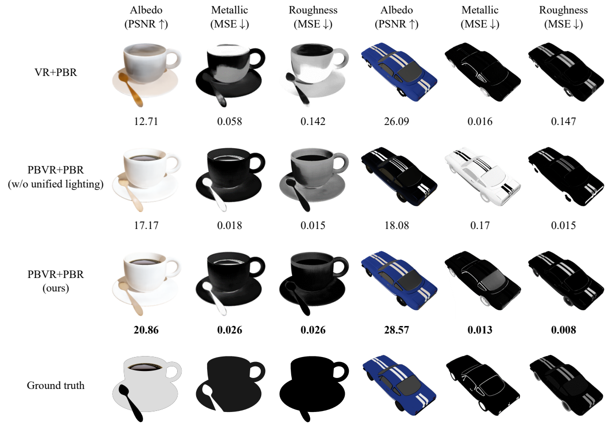

Figure 7: *Ablation study on “coffee” and “car” dataset from ShinyBlender. “VR + PBR” indicates integrating volume rendering and physically based rendering. “PBVR + PBR (w/o unified lighting)” indicates that we used two different light MLPs for PBVR and PBR, respectively. The number below each image indicates the evaluation metric.*

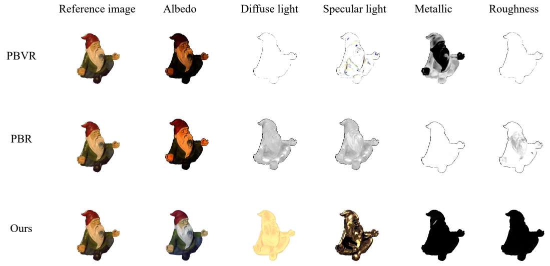

Figure 8: *Comparison with the separate use of PBVR and PBR on the “gnome” from the real-captured data Stanford-ORB demonstrates that our method effectively disentangles lighting and albedo from the appearance.*

Geometry Reconstruction. The evaluation metric used is the Chamfer Distance, provided by the DTU evaluation metrics [AJV*16]. This metric comprises two components: *accuracy* and *completeness*. Consistent with Ref-NeuS [GHZ*23], only accuracy is reported on ShinyBlender and CompoBlender. We also reported the results of Stanford-OBR in the same scale.

Materials Estimation. Given access to ground truth of albedo, roughness, and metallic maps for the synthetic datasets, Mean Squared Error (MSE) was reported for metallic and roughness, and PSNR was used for diffuse albedo. For the real dataset Stanford-ORB, where ground truth for roughness and metallic maps is unavailable, qualitative relighting results including PSNR, SSIM and LPIPS were provided as an alternative. Besides, pseudo albedo was used to evaluate predicted albedo.

Lighting Estimation. For lighting evaluation, akin to DeepLight [LMF*19] and StyleLight [WYLL22], we employ three spheres with different materials for assessment: mirror silver, matte silver, and diffuse grey, depicted in Figure 3. The three spheres are rendered with ground-truth lighting and the estimated environment map using Blender [Hes13]. Evaluation metrics include RMSE, scale-invariant RMSE (si-RMSE) and Angular Error.

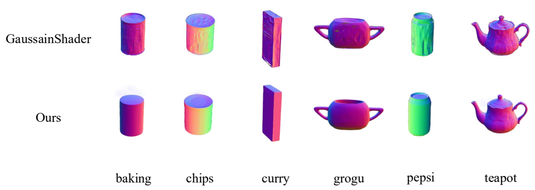

Figure 9: *The surface normal of GaussainShader and our method on Stanford-ORB dataset.*

4.2. Implementation Details

Our model was developed based on NeRO [LWL*23]. The architecture of the geometry network, lighting network, and material network mirrors that of NeRO. For more details, please see

Proceedings published by Eurographics - The European Association for Computer Graphics.

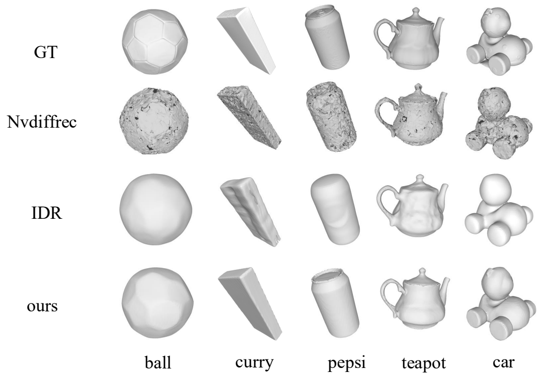

Figure 10: *More reconstruction results on objects from Stanford-ORB. IDR, Nvdffrec are compared.*

our Supplement. Our model underwent training for 200,000 iterations, requiring 10 hours on a single NVIDIA RTX 3090 Ti GPU. Upon convergence, a mesh was extracted from the signed distance functions within a predefined bounding box using the Marching Cubes [LC87] at a resolution of 512. An environment map with a resolution of 512 × 1024 was generated by uniformly sampling across azimuth and elevation in spherical space, followed by querying the light using the direct light MLP. Note that although our approach builds upon NeRO [LWL*23], we believe it can be adapted to any volumetric neural implicit framework. For example, techniques such as Instant-NGP [MESK22] and CUDA-based Monte Carlo sampling can be readily leveraged for acceleration.

4.3. Comparison with State-of-the-Art Methods

We compared the results of our method with several other methods, including NeRO [LWL*23], InvRender [ZSH*22] and NvdiffRec [MHS*22] and a 3D Gaussian Splatting based inverse rendering method GaussianShader (GShader) [JTL*24] on both synthetic dataset, and compared with cutting-edge method NeRO on real captured dataset. The quantitative results are shown in Tables 1 and 2. Since InvRender assume dielectric materials, the metallic and diffuse albedo are not available. We reported the mean result for each evaluation metric. For lighting, we further averaged the results on three spheres. Please refer to our Supplement for more details. Our method significantly outperforms all other compared methods on all evaluation metrics. We shown the qualitative comparison of lighting estimation in Figure 4 and the extracted 2D environment map by querying the optimized direct light MLP in Figure 5. We also visualized the qualitative comparison of geometry reconstruction in Figure 6. Note that though the 3D Gaussian Splatting-based method [JTL*24] excels in optimization speed, its performance is significantly inferior. More visualizations are in the Supplement, where we also discuss how incorrect materials can lead to wrong geometry reconstruction.

Table 3: *Comparison of novel view synthesis with Ref-NeRF and NeRO, including the first stage PBVR and the second stage PBR.* ***Bold*** *results have the best score.*

Methods	PSNR ↑	SSIM ↑	LPIPS ↓
Ref-NeRF	27.86	0.878	0.375
NeRO (PBVR)	29.73	**0.904**	0.326
NeRO (PBR)	27.53	0.866	0.384
Ours (PBVR)	**29.77**	0.902	**0.324**
Ours (PBR)	29.51	0.894	0.329

4.4. More results

Geometry reconstruction results. We show the qualitative comparison with GaussianShader on Stanford-ORB in Figure 9 and geometry comparison with NvdiffRec and IDR in Figure 10.

Novel-view synthesis quality. To show the quality of novel view synthesis (NVS), we additionally reported the NVS quality in Table 3 compared to Ref-NeRF, the PBVR and PBR in NeRO and our method in terms of PSNR, SSIM, and LPIPS. The qualitative comparison with NeRO are visualized in Figure 11. The rendered image of "NeRO (PBR)" is inferior to the same PBR rendered image in our unified framework, which shows that our method enhances PBR anti-aliasing capability. The rendered results of "NeRO (PBVR)" and our framework are comparable.

Relighting quality. For the relighting evaluation, we selected challenging objects that include inter-reflections. Specifically, the "cat" from GlossyBlender dataset and the "coffee" from ShinyBlender dataset were chosen. The results of these evaluations are reported in Figure 12.

5. Ablation Study

We conducted an ablation study on the “coffee” object from ShinyBlender and the “gnome” object from Stanford-ORB to evaluate the effectiveness of unifying PBR and PBVR, as well as the unified light representation. We first compared our method with naively integrating traditional volume rendering and PBR. However, traditional volume rendering struggles to reconstruct reflective surfaces accurately, which is essential for optimizing materials and lighting. Consequently, we employed Ref-NeuS [GHZ*23] for volume ren-

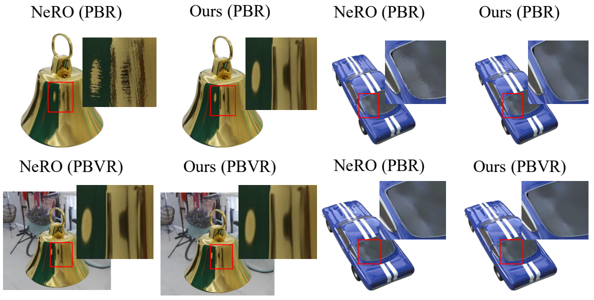

Figure 11: *The extracted environment map from the direct light MLP. Our method can recover fine-grained environment map given only multi-view 2D images.*

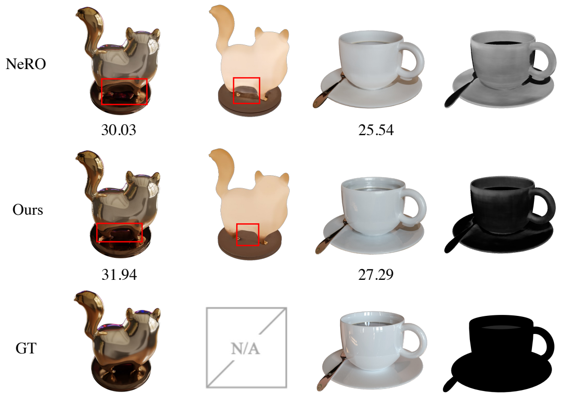

Figure 12: *Relighting comparison with NeRO on the ShinyBlender and Glossy-Blender datasets. Our method excels at accurately estimating materials, including albedo (highlighted in the red box in "cat") and roughness ("coffee"), particularly in scenarios with inter-reflection. Correctly recovering these parameters is crucial for accurate relighting.*

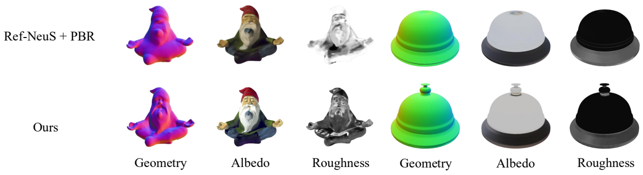

Figure 13: *Ablation study on the Stanford-ORB dataset and Glossy-Blender dataset to validate the effectiveness of combining PBVR and PBR for imposing mutual constraints in inverse rendering.*

dering, which demonstrated ideal reconstruction results for specular surfaces. The results are shown in Figure 7 denoted as "VR + PBR". Next, we highlight the importance of unifying lighting representation by encoding the light for PBVR and PBR with two different light MLPs, respectively. The results are shown in Figure 7 indicated as "PBVR + PBR (w/o unified lighting)". Our method significantly improves the performance of materials estimation. We then compared our method with the two-stage optimization approach on the "gnome" object from the Stanford-ORB dataset to validate the effectiveness of unifying PBR and PBVR. In the first stage, PBVR was utilized for surface reconstruction. In the second stage, based on the derived geometry, PBR was applied for materials and lighting estimation. The results, as illustrated in Figure 8, demonstrate that our method effectively disentangles lighting and albedo from appearance, whereas the two-stage approach with PBVR and PBR results in entangled outputs. More visualization of the ablation study can be found in Figure 13.

5.1. Conclusions

In this paper, we explore the issue of inverse rendering, a topic that serves as a critical bridge between NeRF-based models and conventional rendering engines, yet remains under-explored. The inherent ambiguity among geometry, materials, and lighting can significantly hinder accurate decomposition. Our method, Uni-IR, effectively addresses this challenge by integrating physically based volume rendering and physically based rendering into a unified framework. Both rendering methods directly reason materials, lighting and geometry. With a carefully designed unified representations for both lighting and materials, our approach impose mutual constraints and achieve significant performance on inverse rendering task.

References

[AJV*16] Aanæs H., Jensen R. R., Vogiatzis G., Tola E., Dahl A. B.: Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision (IJCV)* (2016). 7

[BBJ*21] Boss M., Braun R., Jampani V., Barron J. T., Liu C., Lensch H.: Nerd: Neural reflectance decomposition from image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)* (2021). 3

[BM14] Barron J. T., Malik J.: Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)* (2014). 3

[BMV*22] Barron J. T., Mildenhall B., Verbin D., Srinivasan P. P., Hedman P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022). 5

[CL96] Curless B., Levoy M.: A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (1996). 2

[CT82] Cook R. L., Torrance K. E.: A reflectance model for computer graphics. *ACM Transactions on Graphics (ToG)* (1982). 4

[FL95] Fua P., Leclerc Y. G.: Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal of Computer Vision (IJCV)* (1995). 2

[FP09] Furukawa Y., Ponce J.: Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)* (2009). 2

[FSV*23] Fan Y., Skorokhodov I., Voynov O., Ignatyev S., Burnaev E., Wonka P., Wang Y.: Factored-neus: Reconstructing surfaces, illumination, and materials of possibly glossy objects. *arXiv preprint arXiv:2305.17929* (2023). 3

[GHZ*23] Ge W., Hu T., Zhao H., Liu S., Chen Y.-C.: Ref-neus: Ambiguity-reduced neural implicit surface learning for multi-view reconstruction with reflection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023). 1, 3, 7, 8

[Gre18] Gregory J.: *Game engine architecture*. AK Peters/CRC Press, 2018. 1

[GYH*20] Gropp A., Yariv L., Haim N., Atzmon M., Lipman Y.: Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099* (2020). 5

[Hes13] Hess R.: *Blender foundations: The essential guide to learning blender 2.5*. Routledge, 2013. 7

[JTL*24] Jiang Y., Tu J., Liu Y., Gao X., Long X., Wang W., Ma Y.: Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024). 8

[Kaj86] Kajiya J. T.: The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques* (1986). 3

[KLR*22] Kopanas G., Leimkühler T., Rainer G., Jambon C., Drettakis G.: Neural point catacaustics for novel-view synthesis of reflections. *ACM Transactions on Graphics (TOG)* (2022). 3

[KVH84] Kajiya J. T., Von Herzen B. P.: Ray tracing volume densities. *ACM SIGGRAPH computer graphics* (1984). 3

[KZY*24] Kuang Z., Zhang Y., Yu H.-X., Agarwala S., Wu E., Wu J., et al.: Stanford-orb: A real-world 3d object inverse rendering benchmark. *Advances in Neural Information Processing Systems (NeurIPS)* (2024). 6

[Las87] Lasseter J.: Principles of traditional animation applied to 3d computer animation. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques* (1987). 1

[LC87] Lorensen W. E., Cline H. E.: Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics* (1987). 8

[LCL*23] Liang R., Chen H., Li C., Chen F., Panneer S., Vijaykumar N.: Envidr: Implicit differentiable renderer with neural environment lighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023). 1, 3

[LJ02] Lewis M., Jacobson J.: Game engines. *Communications of the ACM* (2002). 1

[LMF*19] LeGendre C., Ma W.-C., Fyffe G., Flynn J., Charbonnel L., Busch J., Debevec P.: Deeplight: Learning illumination for unconstrained mobile mixed reality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). 7

[LWL*23] Liu Y., Wang P., Lin C., Long X., Wang J., Liu L., Komura T., Wang W.: Nero: Neural geometry and brdf reconstruction of reflective objects from multiview images. In *SIGGRAPH* (2023). 1, 2, 3, 4, 7, 8

[LYL*23] Liang Y., Yang X., Lin J., Li H., Xu X., Chen Y.: Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. *arXiv preprint arXiv:2311.11284* (2023). 3

[Max95] Max N.: Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics (VCG)* (1995). 3

[MESK22] Müller T., Evans A., Schied C., Keller A.: Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)* (2022). 8

[MHS*22] Munkberg J., Hasselgren J., Shen T., Gao J., Chen W., Evans A., Müller T., Fidler S.: Extracting triangular 3d models, materials, and lighting from images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022). 8

[MON*19] Mescheder L., Oechsle M., Niemeyer M., Nowozin S., Geiger A.: Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (2019). 2

[MST*21] Mildenhall B., Srinivasan P. P., Tancik M., Barron J. T., Ramamoorthi R., Ng R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* (2021). 1, 2, 3, 5

[NDVZJ19] Nimier-David M., Vicini D., Zeltner T., Jakob W.: Mitsuba 2: A retargetable forward and inverse renderer. *ACM Transactions on Graphics (TOG)* (2019). 3

[Nic65] Nicodemus F. E.: Directional reflectance and emissivity of an opaque surface. *Applied optics* (1965). 1

[OPG21] Oechsle M., Peng S., Geiger A.: Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2021). 1, 2

[Par12] Parent R.: *Computer animation: algorithms and techniques*. Newnes, 2012. 1

[PFS*19] Park J. J., Florence P., Straub J., Newcombe R., Lovegrove S.: Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (2019). 2

[PNM*20] Peng S., Niemeyer M., Mescheder L., Pollefeys M., Geiger A.: Convolutional occupancy networks. In *European Conference on Computer Vision (ECCV)* (2020). 2

[RPHD20] Rodriguez S., Prakash S., Hedman P., Drettakis G.: Image-based rendering of cars using semantic labels and approximate reflection flow. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* (2020). 3

[SCL*23] Sun C., Cai G., Li Z., Yan K., Zhang C., Marshall C., Huang J.-B., Zhao S., Dong Z.: Neural-pbir reconstruction of shape, material, and illumination. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023). 1

[SDZ*21] Srinivasan P. P., Deng B., Zhang X., Tancik M., Mildenhall B., Barron J. T.: 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). 3

[SF16] Schonberger J. L., Frahm J.-M.: Structure-from-motion revisited. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (2016). 2

[SVDSKVDM01] Schuemie M. J., Van Der Straaten P., Krijn M., Van Der Mast C. A.: Research on presence in virtual reality: A survey. *Cyberpsychology & behavior* (2001). 1

[SZFP16] Schönberger J. L., Zheng E., Frahm J.-M., Pollefeys M.: Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision (ECCV)* (2016). 2

[VHM*22] Verbin D., Hedman P., Mildenhall B., Zickler T., Barron J. T., Srinivasan P. P.: Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022). 1, 2, 3, 5, 6

[WHZL24] Wang H., Hu W., Zhu L., Lau R. W.: Inverse rendering of glossy objects via the neural plenoptic function and radiance fields. *arXiv preprint arXiv:2403.16224* (2024). 3

[WLL*21] Wang P., Liu L., Liu Y., Theobalt C., Komura T., Wang W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689* (2021). 1, 2, 3

[WYLL22] Wang G., Yang Y., Loy C. C., Liu Z.: Stylelight: Hdr panorama generation for lighting estimation and editing. In *European Conference on Computer Vision (ECCV)* (2022). 7

[XT19] Xu Q., Tao W.: Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). 2

[YDMH99] Yu Y., Debevec P., Malik J., Hawkins T.: Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (1999). 3

[YGKL21] Yariv L., Gu J., Kasten Y., Lipman Y.: Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems (NeurIPS)* (2021). 1, 2

[YZL*22] Yao Y., Zhang J., Liu J., Qu Y., Fang T., McKinnon D., Tsin Y., Quan L.: Neilf: Neural incident light field for physically-based material estimation. In *European Conference on Computer Vision (ECCV)* (2022). 3

[ZLW*21] Zhang K., Luan F., Wang Q., Bala K., Snavely N.: Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 1, 3

[ZSD*21] Zhang X., Srinivasan P. P., Deng B., Debevec P., Freeman W. T., Barron J. T.: Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)* (2021). 1, 3, 6

[ZSH*22] Zhang Y., Sun J., He X., Fu H., Jia R., Zhou X.: Modeling indirect illumination for inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022). 1, 3, 8

Proceedings published by Eurographics - The European Association for Computer Graphics.